DM 14,-

## DAS GROSSE HEIMCOMPUTER-MAGAZIN

SOFOHUK LIL 📽 📽 👀 Z X C V B N M E 2 2 3 1 me e

Alles für 800 XL und 130 XE:

Tips, Tricks und tolle Listings

<u>Zum Abtippen:</u> Das schnellste

Basic

interpreter und Compiler

## <u>Grundlagen</u>

- Musik
- ★ Compiler
  ★ Einsprungadressen

<u>Basteleien</u>

Hardware- und Software-Tests



#### Das Angebot dieser Sonderausgabe:

Wie auch zu unserer Stammzeitschrift Happy-Computer bieten wir wieder alle in die em Sonderheft veröffentlichten Programme auf Diskette an. Da wir aber lle Programme auf einer Diskette unterbringen konnten, haben wir sie ni Disketten im normalen DOS-2.0-Format verteilt. Somit kann man also as alte 810-Diskettenlaufwerk verwenden. Übrigens ist Diskette 1 beid-11. «uc

Diskette 2 einseing bespielt. Da sich auf allen drei Diskettenseiten ein DOS.SYS-File befindet, können Sie die Disketten booten. Dazu müsnur die entsprechende Diskette ins Laufwerk schieben und die Stromgung des Computers einschalten

Sie irgendwelche Programme ausprobieren, sollten Sie sich Sicherheits-n anlertigen. Am einfachsten und schnellsten geschieht dies mit der action »J» vom DOS-Menü aus.

enn Sie die Diskette I (mit dem Aufkleber nach oben) booten, gelangen Sie automatisch ins Turbo-Basic XL. Mit DIR erfolgt dann die Ausgabe des Inhaltsverzeichnisses auf dem Bildschirm. Sollten Sie noch den alteren Atari 800 ver-wenden, booten Sie bitte die zweite Selte von Diskette 1. Im Atari-Sonderheit finden Sie Programme in normalem Atari-Basic, in Turbo-

Basic XL, in Maschinensprache und in Assembler. Damit Sie die Programme auf Diskette besser zuordnen können, haben wir für die oben genannten Krite-rien jeweils einen bestimmten Extender gewählt. Sie lauten im einzelnen:

XXXXXXX.BAS Atari-Basic. Um solche Programme laden zu können, müssen Sie sich entweder in Atari-Basic oder in Turbo-Basic XL befinden. Achtungt Manche Basic-Programme können unter Turbo-Basic XL zu schnell ablaufen.

XXXXXXX.TUR Turbo-Basic XL. Solche Programme müssen unbedingt unter Turbo-Basic XL laufen. Da dieses Basic über einen erweiterten Befehls-satz verfügt, ist es nicht kompatibel zum Atari-Basic. Sollten Sie dennoch ein sol-

ches Programm mit dem Standard-Atari-Basic laufen lassen, erhalten Sie entsprechende Fehlermeldungen.

XXXXXXX.M65 MAC/65-Quelitexte. Solche Dateien müssen unbedingt mit dem MAC/65-Assembler assembliert werden. Mit Basic lassen sie sich nicht verarbeiten.

XXXXXXXCOM Objekt-Code. Hierbei handelt es sich um Programme, die man sonst mit AMPEL (Atari-Maschinen-Programm-Eingabe-Listing) einen muß. Um solche Programme zu starten, wählen Sie im DOS-Menü die Funktion Le und geben dann den entsprechenden Programmnamen ein. Anschließend wird das Programm geladen und automatisch gestartet. Wenn Sie sich in Turbo-Basic XL befinden, lassen sich solche Programme auch mit \*BRUN\*D:XXXXXXXXCOM« aufrufen und anschließend starten

\*\*\*\*\*\*\*\*\*\*\* System-Dateien (DOS 2.5). Auf der Rückseite von Diskette 1 befindet sich außerdem noch ein DOS.5YS-File mit 145 Sektoren Länge. Hierbei handelt es sich um die abgespeckte Turbo-Basic XL-Version für den Atari 400 oder 800 mit mindestens 48 KByte-RAM. Wenn Sie diese Diskette booten, gelangen Sie automatisch ins Turbo-Basic.

Damit wären also die einzelnen Programmtypen erklärt. Sollten Sie Probleme mit den Programmen auf den Leserservice-Disketten haben, empfiehlt es sich, die entsprechenden Erläuterungen im Sonderheft genau und aufmerksam durchzulesen. So kann manch ein Problem gelöst werden,

Bestell-Nr. LH 86S2D, DM 34,90\*, (2 Disketten)

\*inki, MwSt Unverbindliche Preisempfehlung

#### Programme aus früheren Ausgaben

DM 29.90\*

#### ATARI 800XL/130XE/800 Tarbo-Basic

Ler schnelle Basic-Interpreter für den Atari. Auf der Diskette befindet sich je eine Version für den Atari 800XL und eine für den Atari 800 mit mindestens 48 KByte-RAM, Aus Ausgabe 12/85.

#### AMPEL

Atarı-Maschinen-Programm-Eingabe-Listing Aus Ausgabe 12/85.

#### Atari-Prüfsummer

Eingabehilfe für alle in Happy-Computer veröffentlichten Basic-Programme.

#### Jumper II

Listing des Monats aus Ausgabe 8/84. um die Geschwindigkeit von Turbo-Basic zu demonstrieren.

#### Magic-Painter

Listing des Monats aus Ausgabe 3/88. Ein Zeichenprogramm, das an Turbo-Basic angepast wurde.

Alle 5 Programme auf einer Diskette für den ATARI 800XL/130XE/800.
Bestell-Nx. LH 8512B, DM 29,90\*

#### ATARI 800XL

#### Prüfsummer

Emgabehilfe für alle in Happy-Computer veröffentlichten Basic-Programme.

#### Gt Uheimer

creen-Editor und 20 fertigen Sze-M Spiel) aus Ausgabe 9/85

#### 24 Sen in Grafikstufe 0 Ro

e für farbige Schrift (Utility), aus AL. be 6/85.

#### Die

Di elp Für die schnelle Rettung (Utility), aus Ausgabe 8/85.

Mit dem Atan auf Ölsuche (Spiel), aus Ausgabe 8/85.

#### Autostart

Basic-Programme automatisch starten (Utility), aus Ausgabe 9/85.

Mehr Speicher mit der 1050 Floppy (Utility), aus Ausgabe 10/85

Alle 7 Programme auf einer Diskette für den Atari 800 XI

Bestell-Nr. LH 8510 B,

#### **ATARI 48 KByte-RAM**

#### Magic Painter

Unser Listing des Monats in der Ausgabe 3/85 ist ein Grafikprogramm, das sich mit anderen Malprogrammen dieser Art durchaus messen kann. Besonders gelungen ist die einfache Bedienung, da man mit dem Joystick sowohl im Haupt- als auch in den Untermenüs sämtliche Punkte anwählen kann. Der elektronische Malkasten verfügt über 16 Menüpunkte und bietet eine Grafikauflösung von 160x96 Pixels.

#### Grafildemo

Alle 256 Farben werden auf dem Bildschirm dargestellt. Eine Farbspielerei. die die hervorragenden Grafikfähig-keiten der Atari-Computer beweist (Rainbow-Effekt). Aus Ausgabe 3/85.

#### Variablen-Dump

Mit diesem Programm können Sie die verwendeten Variablen eines anderen Programms auf dem Bildschirm listen. Ein wichtiges Utility, das Ihnen die lästige Fehlersuche in längeren Basic-Programmen erleichtert. Aus Ausgabe

#### Wie die Bilder laufen lernen

Mit dem Utility »Power-Mover« können Sie laufende Bilder schnell und problemlos erzeugen. Für alle, die sich an die Programmierung von Player-Missile-Grafiken heranwagen. Ausgabe 2/85

#### Statuszeile mit Uhr

Damit Sie beim Programmieren nicht die Zeit vergessen, hilft nur eine ständig sichtbare Zeitanzeige. Mit diesem Programm können Sie eine zusätzliche Statuszeile oberhalb des Bildschirms generieren. Aus Ausgabe 1/85

Alle 5 Programme auf Diskette für Atari mit mindestens 48 KByte-RAM.

Bestell-Nr. LH 8503 B, DM 29.90\*

#### ATARI 48 KByte-RAM Diamantenfieber

Unser Listing des Monats aus der Ausgabe 2/85 ist eine wahre Schatztruhe Bereichern Sie sich an bunt glitzernden Diamanten, die kunterbunt in einem Bergwerk verteilt sind. Aber Vorsicht Die Stollen sind sehr unstabil. Eine falsche Bewegung genügt, und Sie wer-den von losen Gesteinsbrocken erschlagen. Ein Spiele-Designer sorgt bei unserer exzellenten »Boulder Dash«-Variante für anhaltende Spannung. Entwerfen Sie Ihre eigenen Bergwerkstollen, aber mit Bedacht, denn nicht selten ergibt sich bei einem Bild nur ein Lösungsweg.

#### Die Schatzhöhle

Wer möchte sich nicht auch mit einem Schatz bereichern. Wer dazu nicht unbedingt eine Weltreise unternehmen möchte, kann mit seinem Atari 800XL in eine Schatzhöhle eindringen. Gefährliche Tiere wie Skorpione, Ratten und Schlangen erschweren die Suche Nur mit viel Geschick können Sie Ihren Geldbeutel – aber nur im Spiel - auffüllen. Aus Ausgabe 1/85.

Die wichtigste RENUMBER Funktion fehlt leider im Standard-Atari-Basic Dieses Programm behebt diesen Mangel. Es ist leicht zu bedienen. Einfach das Basic-Programm von der Diskette laden, «RUN» eingeben, und das entsprechende Objekt-File wird auf Dis-kette geschnieben. Dann nur noch vom DOS-Menü aus mit der Funktion »Le das Objekt File laden und mit Bezurück ins Basic gehen. Wenn Sie ein Basic-Programm umnumerieren möchten, geben Sie einfach PRINT USR (8044) ein, und Ihr Programm wird wunschge mäß umnumeriert. Aus Ausgabe 11/84.

#### Jumper II

Ein professionell gemachtes, in Basic geschriebenes Spiel. Exzellente Pro-grammierung, ein Highscore-Zähler und eingebautes Demo werden selbst Zweifler schnell überzeugen, Auf musikalische Untermalung wurde großer Wert gelegt, und die verschiedenen Screens sind brillant gemacht. Genau das richtige Programm für kalte Win-terabende Listing des Monats. Aus Ausgabe 8/84,

Mop – Der Goldgräber Schnelligkeit und guter Sound zeich-nen dieses Spiel aus. Die Soundfähig-keiten sind wirklich hervorragend. Viele Bilder sorgen bei diesem Programm für viel Abwechslung. Ausgabe 7/84.

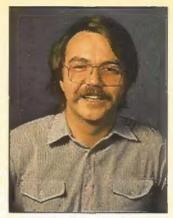
Alle 5 Programme auf Diskette für Atari mit mindestens 48 KByte-RAM

Bestell-Nr. LH 8502 B, DM 29,90\*

Bestellungen aus der Schweiz richten Sie bitte direkt an: Markt & Technik Vertriebs AC, Kollerstr. 3, CH-6300 Zug. Tel. 042/415656.
Bestellungen aus anderen Ländern bitte per Auslandspostanweisung! Achtung; Nicht die eingeheftete Zahlkarte verwenden!
Bestellungen aus Österreich richten Sie bitte direkt an: Ueberreuter Media Handels- und Verlagsges. mbH, Alser Str. 24, 1091 Wien, Tel. 0222/481538-0

<sup>\*</sup> Alle Preise inklusive Mehrwertsteuer, unverbindliche Preisempfehlung, Leser-Service-Produkte sind nur für Endkunder, nicht für Wiederverkäufer





Is Atari-Besitzer kommt man sich manchmal wie auf einer einsamen Südsee-Insel vor. Sucht man Informationen oder Listings für seinen 800XL oder 130XE in deutschen Computerzeitschriften, sitzt man leider allzuoft auf dem Trockenen. Diesem Notstand wollen wir mit diesem Sonderheft abhelfen. Hier finden Sie eine Fülle von Listings und Informationen, die Sie hoffentlich für die lange Durststrecke entschädigen. Der Atari 800XL/130XE zählt schließlich nach wie vor zu den leistungsfähigsten derzeit erhältlichen Heimcomputern. Bis zu 256 gleichzeitig darstellbare Farbschattierungen und die Player Missile-Grafik, mit der sich einige Effekte viel besser umsetzen lassen als mit Sprites, sind nicht zu übersehen. Berücksichtigt man dann den sehr günstigen Anschaffungspreis des Atari 800XL/130XE, ist dessen Preis/Leistungsverhältnis wirklich von kaum einem Heimcomputer zu schlagen. Auch die Peripherie, also Disketten-Laufwerke und Drucker, ist in letzter Zeit preiswert geworden.

Da wären aber noch die neuen 16-Bit-Computer. Man weiß, daß der Atari 800XL und der 130XE mit einer 8-Bit-CPU vom Typ 6502 ausgestattet ist. Damit scheint klar zu sein, daß ein 16-Bit-Prozessor viel leistungsfähiger sein muß als das 8-Bit-Gegenstück. Zählt also der 800XL im Vergleich zum ST zum alten Eisen? Sicher kann der 800XL nicht mit der Grafikauflösung oder der Geschwindigkeit des ST konkurrieren. Aber einen ganz großen Pluspunkt gibt es für den 800XL: nämlich Software für jeden Zweck. Als Spielecomputer zum Beispiel, ist der ST zur Zeit noch keine Alternative, Natürlich wird es in Zukunft viel Software für diesen Computer geben, aber wer sofort loslegen möchte und vorzugsweise auf fertige Programme Wert legt, ist mit dem 800XL besser bedient.

Als wirklich hervorragendes Beispiel für ausgezeichnete Software mag unser Turbo-Basic-Interpreter mit zugehörigem Compiler dienen. Beide Pro-

gramme finden Sie in diesem Sonderheft - natürlich zum Abtippen. Sie brauchen also nur ein wenig Zeit zu investieren und haben anschließend einen Interpreter und einen Compiler, die es in sich haben. Geschwindigkeitsvergleiche haben nämlich ergeben, daß Turbo-Basic XL seinem Namen alle Ehre macht; dieser Interpreter zählt zu den schnellsten, die es für den Atari gibt. Der dazu passende Compiler compiliert sogar in normalem Basic geschriebene Programme. Diese Programme sind dann wirklich die absoluten »Renner«. Und damit Sle Ihr Turbo-Basic XL auch gleich ausprobieren können, haben wir einige Programme in dieser Programmiersprache veröffentlicht.

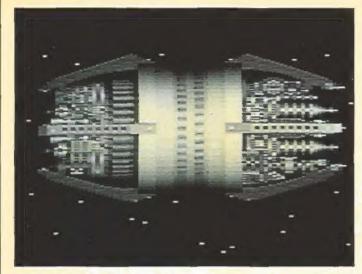
Warum sollte man also neidisch werden, wenn in den meisten Zeitschriften mehr über andere Computer berichtet wird. Schließlich wird man einige Zeit brauchen, bis man alle in diesem Sonderheft veröffentlichten Programme ausprobiert hat. Oder bauen Sie sich doch die eine oder andere Hardwarebastelei nach. Sei es eine Zehnertastatur oder ein Cartridge-Experimentier-System, das Angebot ist reichhaltig und sorgt sicher für einige unterhaltsame Stunden.

Ebenfalls in diesem Sonderheft finden Sie eine Umfrage, bei der es eine Menge schöner Preise als Belohnung fürs Mitmachen zu gewinnen gibt. Wir möchten nämlich gerne Ihre Meinung zu diesem Sonderheft wissen. Schließlich brauchen wir Ihre Anregungen und Vorschläge, um weitere Sonderhefte und natürlich vor allem unsere Stammzeitschrift Happy-Computer für Sie so interessant wie möglich zu gestalten. Unter allen Einsendungen verlosen wir unter anderem zwei Atari 1050 Disketten-Laufwerke und einen Atari 1029 Nadel-Matrixdrucker. Also nicht lange zögern! Fragebogen ausfüllen, in ein Kuvert stecken und ab die Post an die Redaktion. Die Gewinner werden dann in einer der nächsten Ausgaben von Happy-Computer bekanntgegeben.

Werner Breuer

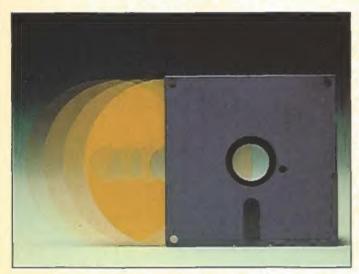
## Atari im Aufwind

### INHALT-



Spiele, die es in sich haben

10



Das Atari 1050-Diskettenlaufwerk wird zur Rennfloppy

#### **Diskettenservice**

Wer keine Zeit oder keine Lust hat, alle Programme selbst in mühevoller Kleinarbeit abzutippen, kann wieder auf den bewährten Diskettenservice zurückgreifen. Es sind hier sämtliche Programme des Sonderhefts auf zwei Disketten erhältlich.

Bestellnummer LH 86 S2D

**29,90 DM** (2 Disketten)

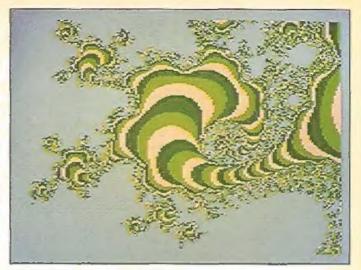
Hardware-Tests	
Auf die Verbindung kommt es an (Schnittstellen)	6
Rasende Daten (»Floppy-Speeder« für das Atari 1050-Laufwerk)	7
Die Raupe wird zum Schmetterling (Speichererweiterung für den 800XL)	8
Basteleien	
Cartridge-Experimente leichtgemacht (Experimentiersystem)	12
PIA - die dritte Schnittstelle (Zusatztastatur am Joystickport)	17
Roboter im Griff (Robotersteuerung)	21
Das schnellste Basic	
Der große Turbo-Basic-Teil	24
So funktioniert der Turbo-Basic-Compiler (Grundlagen)	25
Der Unterschied liegt im Detail (Kompatibilität zwischen Atari-Basic und Turbo-Basic XL)	30
Auf die Taste, fertig, los! (Geschwindigkeitsvergleich)	32
Zum Abtippen:	
Interpreter	34
Compiler	48
Von der Ordnung zum Chaos-Apfelmännchen auf dem Atari	61
Grafikspielereien in Turbo-Basic XL.	67
Mehr Tempo für Player Missiles	69
Daten komprimiert gespeichert	70
baten compramer geopotorer	
Anwendungs-Listing	
Ordnung muß sein (Disksorter)	71
Spiele-Listings	
Submission – ein Weg voller Gefahren (Labyrinth-Spiel)	75
Vorsicht Falle (Aktion-Spiel)	81
Pacman mal 2	
(Pacman-Variante für zwei Personen)	84





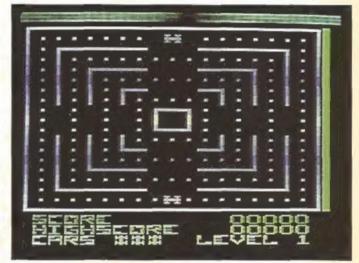
### SONDERHEFT 2186

Einenhahilfan	
Eingabehilfen  AMPEL-Version 1.1	87
Prüfsummer paßt auf	90
Tips&Tricks	
Farbe auf die Fläche (Filfroutine)	93
Mit »Happy-Mon« auf der Suche (Monitor)	96
Windows: Nicht nur ein Augenschmaus (Windowroutine)	103
Basic-Schalter (Basic ein- oder ausschalten)	110
Blitzschnelle Zeichenumwandlung (Zeichensatzumwandlung)	110
Bildschirmausschnitt schnell gelöscht (Löschroutine)	112
Daten schnell zur Hand (Schnelladeroutine)	113
Wortumbruch perfekt (Ausgaberoutine)	114
PEEKs und POKEs mit List und Tücke (Wichtige Speicheradressen)	118
Grundlagen	
Compiler	25
Einsprungadressen	115
Musik	121
Software-Tests	
Spielmaschine Atari	10
Programmiersprache: Aktion mit Action	131
Das Textverarbeitungs-Sextett	133
150 Befehle mit Basic XE	138
DOS-Parade	139
MAC/65: ein Assembler sprintet los!	142
»SynFile+« Dateiverwaltung total	144
Rubriken	
Einleitung	3
Wettbewerb	106
Bücher	108



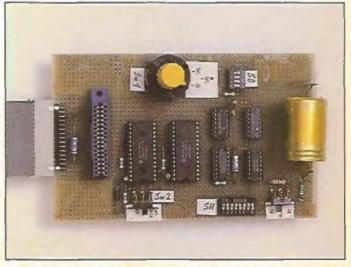
Bunt und doch immer anders: Apfelmännchen

61



Mit 200 um die Ecke düsen, aber »Vorsicht Falle!«

81



Cartridges selbst unter die Lupe nehmen

12



## Auf die Verbindung kommt es an

Schnittstellen stellen die Verbindung zwischen Computer und Außenweit dar. Wir haben zwei Interfaces unter die Lupe genommen.

ill man seine Programme nicht nur auf dem Bildschirm bewundern, so benötigt man einen Drucker und daneben auch eine passende Schnittstelle. Beim Atari kann man dieses Problem auf zwei Arten lösen. Besitzt man einen Original-Atari-Drucker, wie zum Beispiel den Atari 1029, so ist keine separate Schnittstelle nötig. Diese Drucker sind nämlich dafür ausgelegt, ihre Daten über den seriellen Bus des Atari zu empfangen. Der Nachteil dabei ist, daß solche Drucker wirklich nur für den Atari verwendungsfähig sind. Bei einem Systemwechsel muß man sich dann zwangsläufig einen neuen Drucker anschaffen.

Die Alternative hierzu ist ein belieblger Drucker mit Centronics-Schnittstelle. Diese Schnittstelle entspricht einer bestimmten Norm und hat die Eigenschaft, Daten parallel zu verarbeiten. Die Datenübertragung des Atari jedoch erfolgt seriell. Daher muß eine Schnittstelle die Wandlung des Datenformats vornehmen. Bis vor etwa zwei Jahren gab es deshalb bei Atari auch das Interface 850 zu kaufen. Neben einer Centronics-Schnittstelle waren hier auch vier serielle Schnittstellen vom Typ RS232 eingebaut. Daran lassen sich vor allem Akustikkoppler und auch einige Drucker anschließen. Dieses Interface ist jedoch nicht mehr erhältlich, und so übernahmen es verschiedene Fremdhersteller, die Schnittstellen für den Atari anzubieten.

Eine dieser Schnittstellen nennt sich Interface 850 XL und kostet rund 400 Mark inklusive Netzteil und Drukkerkabel. Neben einer Centronics-Schnittstelle ist zwar nur eine RS232-Schnittstelle eingebaut, diese genügen allerdings für die meisten Anwendun-

gen, Der RS232-Teil des Interfaces ist voll kompatibel zur alten Atari-850-Schnittstelle. Den Centronics-Teil jedoch hat man etwas erweitert. Ein Nachteil der 850-Schnittstelle gegenüber der Verwendung der Original-Atari-Drucker war nämlich die fehlende Codewandlung der Grafikzeichen. Nicht jeder Drucker, der an die Centronics-Schnittstelle angeschlossen werden kann, ist auch in der Lage, Grafikzeichen zu drucken (zum Beispiel Typenraddrucker). Darum werden alle Codes unverändert an den jeweiligen Drucker geschickt. Dieser Nachteil läßt sich nur durch entsprechende Treibersoftware wieder beheben.

Beim Interface 850 XL hat man nun versucht, diesen Nachteil direkt mit der Schnittstelle wieder wettzumachen. Das Interface ist in vier verschiedenen Druckmodi zu betreiben, die über eine kurze Steuersequenz ausgewählt werden. Im ersten Modus werden alle Zeichen ohne Codewandlung an den Drukker geschickt. Im zweiten Modus, der standardmäßig eingestellt ist und dem 850-Interface-Modus entspricht, wird nur der »Carriage Return«-Code umgewandelt. Der Drucker benötigt zum Zeilenvorschub nämlich den Code 13, während der Atari dazu den Code 155 verwendet. Der dritte Druckmodus ist für Drucker gedacht, die nicht grafikfähig sind. Alle Codes, die beim Atari mit Grafikzeichen belegt sind, werden einfach in Standard-ASCII-Codes umgewandelt. Druckt man nun beispielsweise ein Listing aus, in dem Grafikzeichen vorkommen, so erscheinen diese zwar nicht in ihrer richtigen Form, der Drucker jedoch interpretiert diese Zeichen wenigstens nicht als Steuercodes. Wie der Zeichensatz des Atari in diesem Modus aussieht, sehen Sie in Bild 1. Der vierte Modus Ist für grafikfähige Drucker gedacht, Mit jedem Grafikzeichen wird auch ein entsprechender Steuercode an den Drucker geschickt, der ihn in den Grafikmodus umschaltet. Leider war man hierbei nicht ganz konsequent. Alle invertierten Zeichen werden nicht invertiert gedruckt (Bild 2). Bei normalen Basic-Listings fällt dies nicht weiter ins Gewicht. Verwendet man aber beispielsweise in einem Programm Maschinenunterroutinen, die als Strings und somit in Form von Grafikzeichen abgelegt sind, enthält das Druckerlisting schlichtweg falsche Informationen.

Neben den beiden Schnittstellen enthält das Interface 850 XL auch noch ein Kassettenrecorder-Interface. Man kann so statt des Atari-Datenrecorders auch jeden anderen handelsüblichen Kassettenrecorder an den Atari anschließen.

Benötigt man keine RS232-Schnittstelle, kann man sich auch ein reines Druckerinterface, also nur mit Centronics-Schnittstelle, kaufen. Zum Beispiel das Atari-Interface 72000 zum Preis von rund 250 Mark. Es benötigt kein Netzteil, da es direkt vom Atari mit Strom versorgt wird. Das Gehäuse ist im Druckerkabel integriert. Mit Hilfe von Dipschaltern kann man auch hier zwei verschiedene Druckmodi einstellen. Der erste Modus schickt, mit Ausdes »Carriage Return«nahme Zeichens, alle Codes unverändert weiter und entspricht somit wieder dem 850-Interface von Atari. Der zweite Modus ist speziell für Besitzer des Atari-Schreibers interessant. Dieses Textverarbeitungsprogramm benutzt recht seltsame Codes für die deutschen Umlaute. Bei manchen Druckern kann es daher zu Schwierigkeiten beim Ausdruck kommen. Durch eine entsprechende Codewandlung hebt dieses Interface diesen Nachteil auf.

Neben diesen beiden Interfaces sind vor allem aus den USA noch eine Reihe von Druckerschnittstellen, die mehr oder weniger dieselben Funktionen ausüben, auf dem Markt. Generell kann man sagen, daß die Anschaffung eines Druckerinterfaces und eines normalen Druckers dem Kauf eines Atari-Druckers vorzuziehen ist. Dies gilt vor allem dann, wenn man plant, sich im Laufe der Zeit einen anderen Computer anzuschaffen.

(Wolfgang Czerny/wb)

Bezugsquellen:

Kunkel Industriebedarf, Zweibrückener Str. 8, 7000 Stuttgart 31, Tel. (0711) 884711

Reinhard Wiesemann, Winckenbechstr, 3a, 5600 Wuppertal 2, Tel. (0202) 505077

+i++ '"#\$%%'()\*+,-./0123456789:;<=>?\$ABCDEFGHIJKLMNOPORSTUVWXYZA6U^\_\*\*abcdefghijklmnopqrstuvwxyz★!K4▶

Bild 1. Beispielsausdruck mit dem 850XL-Interface. In Modus 3 werden sämtliche Steuer- und Grafikzeichen in ASCII-Zeichen umgewandelt.

Bild 2. In Modus 4 lassen sich auf einem grafikfähigen Drucker auch Steuer- und Grafikzeichen wiedergeben.



## Rasende Daten

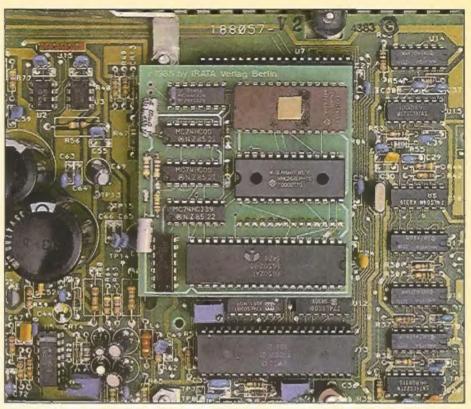
Die Erweiterung »High Speed 1050« macht ihrem Laufwerk Beine und erhöht auch noch die Speicherkapazität.

Floppy-Speeder Speed 1050« für das Atari 1050-Laufwerk besteht lediglich aus einer kleinen Erweiterungsplatine. Auf dieser Platine befinden sich neben einigen Gatterbausteinen auch ein neuer EPROM, ein RAM-Baustein und ein Prozessor vom Typ 6502. Zum Einbau der Erweiterung muß man als erstes das Gehäuse der Diskettenstation öffnen. Dann hebt man vorsichtig das Laufwerk aus seiner Halterung. Nun liegt die Hauptplatine, auf der sich der alte ROM- und der alte RAM-Baustein befinden, offen da. Diese beiden ICs werden aus ihren Sockeln gezogen und statt dessen die Erweiterungsplatine in eine der frei gewordenen Fassungen gesteckt. Der Einbau ist hiermit bereits beendet. Es versteht sich von selbst. daß man dabei größte Vorsicht walten lassen muß. Der Garantieanspruch erlischt nämlich, sobald die Abschirmung der Hauptolatine entfernt wurde.

#### Schnell wie der Wind

Ein auf diese Weise umgebautes Laufwerk hat nun mehrere Vorteile. Zunächst sorgt das neue EPROM dafür, daß man Disketten nun auch mit doppelter Dichte formatieren kann. Das normale 1050er-Laufwerk kann ja neben der einfachen Schreibdichte bereits mit erhöhter Schreibdichte formatieren. Bei der einfachen Dichte wird mit 720 Sektoren von je 125 Byte gearbeitet. Das ergibt eine Speicherkapazität von 88 KByte. Bei der erhöhten Schreibdichte steigt die Zahl der Sektoren auf 1040 an, pro Sektor sind aber immer noch 125 Byte vorgesehen. Die Speicherkapazität ist also auf 126 KByte gestiegen, Doppelte Dichte bedeutet nun, daß man zwar wieder nur über 720 Sektoren, diesmal jedoch von je 256 Byte und somit über 180 KByte Speicherplatz verfügt.

Die Formatierung auf doppelte Schreibdichte erfordert natürlich auch ein neues DOS. Aus diesem Grund bekommt man zusammen mit der Erweiterungsplatine auch gleich eine Diskette mit zwei verschiedenen DOS-Versionen mitgeliefert. Bei »Warp-Speed-DOS« handelt es sich um ein



Acht Bauteile verwandeln ein 1050-Laufwerk in eine Rennfloppy

erweitertes DOS 2.0, das wahlweise Disketten mit einfacher oder doppelter Dichte verarbeitet. Die andere DOS-Version mit dem Namen »MYDOS« ist auch interessant. Hiermit kann man auch wieder unterschiedliche Schreibdichten formatieren. Daneben lassen sich beispielsweise aber auch Subdirectories anlegen.

Alle Daten, die gespeichert oder geladen werden, werden in dem neu hinzugekommenen RAM-Baustein zwischengespeichert. Ein Test ergab, daß sich bis zu 1875 Byte in diesem RAM unterbringen lassen. Speziell bei Dateiverwaltungsprogrammen kann es daher vorkommen, daß Datensätze von Diskette gelesen werden, ohne daß das Laufwerk tatsächlich auf die Diskette zugreifen muß. Der Schreib-/Lesekopf wird also bei Programmen, die häufigen Diskettenzugriff erfordern, geschont. Andererseits hat diese Sache auch einen Haken. Bearbeitet man eine Datei und schließt man den einen Schreibvorgang nicht korrekt mit CLOSE ab, so bleiben die Diskettenanweisungen im Puffer. Wechselt man nun die Diskette im Laufwerk, kann es passieren, daß die im Puffer verbliebenen Anweisungen auf die neue Diskette geschrieben werden und dort Files zerstören.

Durch diese Erweiterung wird das Laufwerk auch um einiges schneller. Vor allem beim Lesen von Daten sind Geschwindigkeitssteigerungen um 50 Prozent keine Seltenheit. Um welchen Faktor sich die Geschwindigkeit im einzelnen erhöht, ist dabei stark vom jeweiligen Lesebefehl abhängig. So verringert sich die Zeit, die das Laufwerk beispielsweise zum Booten des DOS.SYS-Files benötigt, nur um rund eine Sekunde. Liest man aber zum Beispiel 1000 Zahlen mit dem INPUT-Befehl, so sind dazu statt rund 35 Sekunden nur mehr rund 20 Sekunden nötig. Das Formatieren einer Diskette beansprucht nur noch 26 Sekunden statt 38 Sekunden. Speichert man Programme mit dem LIST-Befehl auf Diskette, so hält das Laufwerk sogar zwischenzeitlich immer wieder an, da die Datenübertragung langsamer ist als das Speichern.

### Formatieren in 26 Sekunden

Diese Geschwindigkeitssteigerung hat allerdings auch einen Nachteil. Manche professionellen Programme lassen sich nicht mehr laden, da ihr Kopierschutz auf die veränderte Zugriffszeit reagiert. Mit geeigneter Software wäre es allerdings möglich, das Laufwerk wieder zu normalisieren. Dann stellt das Laden dieser Programme mit dem erweiterten Laufwerk auch kein Problem mehr dar.

Die rund 230 Mark, die man für die Erweiterung ausgeben muß, sparen also nicht nur die Hälfte an Disketten ein, sondern bieten zusätzlich ein schnelleres Laufwerk. Somit ist der Einbau durchaus zu empfehlen.

(Wolfgang Czerny/wb)

Bezugsquelle: IRATA, Hermannstr, 9, 1000 Berlin 44, Tel.: (030) 6212071



## Die Raupe wird zum Schmetterling

eine Erweiterungsplatine, durch die der 800XL voll kompatibel zum 130XE wird. Um es allerdings gleich vorweg zu sagen: Wer Veränderungen an der Hardware seines Atari 800XL vornimmt, riskiert natürlich den Verlust des Garantieanspruchs. Man sollte den Einbau dieser Platine also entweder dem Fachmann überlassen oder aber entsprechende Grundkenntnisse mitbringen.

Auf der neuen Platine befinden sich neben einigen Gatterbausteinen sechs zusätzliche RAM-Bausteine. Die Erweiterung hat ihren Platz direkt auf der Hauptplatine des Atari 800XL Das Gehäuse muß also geöffnet und das Abschirmblech entfernt werden. Die alten RAM-Bausteine werden aus ihren Sockeln gezogen und in die dafür vorgesehenen Sockel auf der Erweiterungsplatine eingesetzt. Die Platine selbst wird in zwei der so freigewordenen Sockel auf der Hauptplatine gesteckt. Nun sind noch einige Drähte von der Erweiterungsplatine an verschiedenen Stellen auf der Hauptplatine anzuschließen. Da Lötarbeiten vermieden werden sollten, schließt man die Drähte direkt an verschiedene ICs an. Dies geschieht, indem man die Drahtenden zusammen mit den IC-Anschlüssen in die Sockel klemmt. Manche Fälle erfordern aber auch ein Hochbiegen der Anschlüsse der ICs, so daß man die Drähte direkt anklemmen kann.

Beim Einbau ist etwas Fingerspitzengefühl unumgänglich. Sonst kann es allzuleicht passieren, daß man versehentlich ein IC-Beinchen abbricht oder das eine oder andere IC durch statische Aufladung zerstört. Aus diesem Grund sollte man vielleicht überlegen, ob man die Erweiterungsplatine nicht doch vom Hersteller der Platine in seinen Atari einbauen läßt.

Ist die Platine an ihrem Platz, verfügt man über einen zusätzlichen RAM-Speicherbereich von 64 KByte. Leider sind diese 64 KByte nicht genauso ansprechbar wie der normale RAM-Speicher. Da der Atari als 8-Bit-Computer nur 64 KByte direkt adressieren kann, muß man den Zusatzspeicher über das sogenannte »Bank-Switching«-Verfahren ansprechen. Gibt man in Basic also beispielsweise den FRE(0)-Befehl ein, hat sich am freien

Um in den Genuß von 64 KByte RAM mehr Speicherplatz zu kommen, muß man sich nicht unbedingt einen Atari 130XE kaufen. Eine Platine verwandelt Ihren 800XL in einen 130XE.

Speicherplatz scheinbar nichts geändert.

Der Zusatzspeicher ist in 4 Banken aufgeteilt, die zu erreichen es einen kleinen Umweg erfordert. Das Bild zeigt, wie die Speicherbereichsverteilung unter 128-KByte-RAM aussieht. Jede der vier Banken umfaßt 16 KByte und muß separat adressiert werden. Diese Aufgabe übernimmt beim 800XL der PIA-Baustein. Dieser Interface-Baustein verfügt über zwei 8 Bit breite Ports. Bei den alten Atari 400 und 800 übernahmen diese beiden Ports die Abfrage der Joysticks. Beim 800XL hingegen wurden zwei Joystickports eingespart. Der dadurch freigewordene zweite Port der PIA wird nun zur Steuerung der MMU (Memory Management Unit oder Speicherverwaltungs-Einheit) eingesetzt. POKEt man also einen entsprechenden Wert in das Portregister, hat man der MMU so mitgeteilt, welche Bank angesprochen werden soll.

#### Speicherriese 800XL

Der Zusatzspeicher eignet sich also weniger zur Aufnahme größerer Basic-Programme, sondern vielmehr dazu, um darin Daten abzulegen; Daten wie zum Beispiel Maschinenunterprogramme oder Player/Missile-Daten. Man kann natürlich auch mehrere Grafik-Bilder

darin ablegen oder ganz einfach allgemeine Datenfelder. Der Vorteil liegt klar auf der Hand, schreibt man ein umfangreiches Programm, kann manches Nachladen von Diskette entfallen. Programme wirken dann nicht mehr »langatmig« und der Anwender wird nicht mehr einer Geduldsprobe unterzogen. Arbeitet man beispielsweise mit dem Atari-DOS 2.5, ist es ohne weiteres möglich, in diesem Speicherbereich eine RAM-Disk anzulegen. Diese RAM-Disk kann dann wie ein Diskettenlaufwerk angesprochen werden. Man speichert oder lädt seine Programme nur mit der Gerätebezeichnung »D8:«. Da beim Laden und Speichern von Programmen die RAM-Disk keine Input/ Output-Operationen, sondern lediglich Speicherverschiebungen durchgeführt werden, gewährleistet dies so natürlich ein viel schnelleres Arbeiten. Befindet sich das DUP.SYS-File beispielsweise in der RAM-Disk, ist man beim Aufruf des DOS-Befehls innerhalb von Sekundenbruchteilen bereits im DOS-Menü.

Der Ausbau eines Atari 800XL auf 128 KByte RAM ist also eine durchaus lohnenswerte Angelegenheit. Lobenswert ist auch die Tatsache, daß der Selbsteinbau sowie der Einbau vom Hersteller das gleiche, nämlich 259 Mark kostet. Sollten Sie also kein erfahrener Bastler sein, ist es ratsam, den Ausbau vom Hersteller vornehmen zu lassen. Schließlich kostet es nicht mehr und eine einwandfreie Funktionsweise des Computers ist auch gewährleistet. Billiger als ein neuer 130XE ist dies auf jeden Fall.

(Wolfgang Czerny/wb)

Bezugsquelle: Wilhelm Bock, Bleichstr. 5, 4790 Paderborn, Tel. (05251) 32691

Adressen	Normaler RAM- und ROM-Bereich	Erweiterter RAM-Bereich
65536-\$FFFF 42192-\$C000	Antic, Pokey, Gtia, Betriebssystem ROMs	7
42191~\$8FFF 32768-\$8000	Basic oder RAM Bildschirmspeicher	
32767-\$7FFF	RAM	Bank 3 \$C000-\$FFFF
		Bank 2 \$8000-\$BFFF
		Bank 1 \$4000-\$7FFF
16384-\$4000		Bank 0 \$0000-\$3FFF
16383-\$3FFF 00000-\$0000	Betriebssystemvariablen usw.	

Die Speicheraufteilung mit 128 KByte-RAM



## ACHTUNG Atari 800XL- und Atari 130XE-Fans!

Das Listing des Monats im März-Heft von »Happy-Computer« bringt: »Happy-DOS II + /D mit RAM-Disk« kompatibel zu DOS 2.0.

Außerdem stellen wir Ihnen drei Monitore für den Betrieb am Atari-ST vor und berichten über das aktuelle Computerge-schehen auf der CES in Las Vegas. Eine große Marktübersicht informiert Sie über das aktuelle Taschencomputerangebot samt zugehöriger Software. In einem Grundlagenbeitrag lernen Sie die logischen Operationen und deren erfolgreiche Anwendung in eigenen Programmen kennen.

Commodore-Fans und -Interessenten finden unter anderem einen Testbericht über das Super-Grafikprogramm »Delu-xe Paint« für den Amiga. Wer sich für Schneider-Computer interessiert, sollte die g Marktübersicht »Hard-Software für Schneider große 464, 664 und 6128« nicht ver-

säumen.





FÜR EIN KOSTENLOSES PROBEEXEMPLAR VON HAPPY COMPUTER

JA, ich möchte «Happy-Computer» kennenlernen.

je, ich mocime shappy-computers zennemernen. Senden Sie mir bitte die aktueliste Ausgabe kostenios als Probeeremplar. Wenn mir «Happy-Computer» gesällt und ich es regelmäßig weiterbeziehen möchte, brauche ich nichts zu tun: Ich erhalte «Happy-Computer» dann regelmäßig frei Haus per Post und bezahle pro Jahr nur DM 66.— statt DM 72.— Einzelvezkaufspreis (Austand auf Anfrage).

Voresme, Name

Straße

PLZ. Ort

Destates

L Unterschrift

Mir ist bekannt, daß ich diese Bestellung innerhalb von 8 Tagen bei der Bestelladresse widern-fen kann und bestätige dies durch meine zweite Unterschrift. Zur Wahrung der Frist genügt die rechtzeitige Absendung des Widerrufs.

2. Unterschrift

Gutschein ausfüllen, ausschneiden, in ein Envert stecken und absenden an: Markt&Technik Verlag Aktiengesellschaft, Vertrieb, Postfach 1304, 8013 Haar



Die Angst des Raumfahrers vor der Untertasse: »Koronis Rift«



»Dimension X« setzt Grafik-Maßstäbe

## Spielmaschine Atari

Daß der Atari bis heute einen Ruf als Spielcomputer genießt, hat seine guten Gründe. Wenn auch Sie mit Ihrem Computer gerne eine Runde spielen möchten, finden Sie hier einige allgemeine Tips.

piele-Freaks, die einen 800XL oder 130XE besitzen, haben momentan allen Grund zu leichtem Frust. Die beiden Atari-Computer sind bekanntlich Sound- und Grafik-Computer allerhöchster Güte. Leider vernachlässigen die Softwarefirmen seit ein paar Jahren die Atari-Heimcomputer. Es erscheinen bei weitem nicht so viele Programme wie für den Marktführer Commodore 64. Hin und wieder gibt es auch Umsetzungen von C 64-Titeln, deren Qualität allerdings mitunter sehr zu wünschen übrigläßt, und die auch die Hardware nicht richtig ausreizen.

#### Frisch aus den USA im Direktimport

Der Atari-Spieler hat kein leichtes Leben, da in letzter Zeit relativ wenig Software-Nachschub anrollt. Bis auf U.S. Gold, die viele Atari-Versionen amerikanischer Programme im Angebot haben, vernachlässigen die englischen Softwarehäuser XL und XE geradezu sträflich. In den USA werden neue Spiele zwar bevorzugt für C 64 und Apple II entwickelt, doch gibt es eine ganze Reihe von Atari-Umsetzungen, die aber hierzulande nur sehr schwer zu erhalten sind. So kann man beim Compy-Shop zum exklusiven Preis von je 149 Mark einige direkt importierte Atari-Programme kaufen, die nur vorerst oder generell nicht in Deutschland vertrieben werden. Auf der jüngsten Angebotsliste findet man hier Leckerbissen wie den neuen U-Boot-Simulator von Microprose, »Silent Services, die Synapse-Adventures »Mindwheels, »Essex« und »Brimstone« sowie das Autorennen »Pitstop II«.

#### »Winter Games« im Anmarsch

Etwas billiger kommen Sportspiel-Fans weg. Zwei der drei erfolgreichen Olympia-Simulationen von Epyx gibt es auch auf Diskette für Atari-Computer: »Summer Games« und »Winter Games« kann man zum erschwinglichen Preis von knapp 60 Mark kaufen. Die Atari-Versionen enthalten die Original-Disziplinen, sind grafisch aber nicht ganz so stark.

Das Programmierteam von Lucasfilm Games hat erst vier Spiele veröffentlicht, doch ein Titel ist besser und erfolgreicher als der andere: »Rescue on Fractalus«, »Ballblazer«, »The Eidolon« und »Koronis Rift«, »Koronis Rift« ist die ausgereifteste und interessanteste Neuerscheinung. Als Raumkapitän düsen Sie mit Ihrer Lieblingsuntertasse durchs All. Plötzlich schlagen alle Instrumente aus: Sie haben Koronis Rift, den sagenumwobenen Planeten einer alten Rasse, wiederentdeckt. Im Raumgleiter rasen Sie über die Planetenoberfläche, um alte Raumschiff-Wracks nach wertvollen Überbleibseln der Supertechnik der verschollenen Ureinwohner zu durchsuchen. Mit diesen Teilen kann man dann das eigene Schiff besser ausrüsten, was auch nötig ist, wenn man den massiven Angriffen der Wächter-Flotten auf Dauer widerstehen will.

#### Neues von den Atari-Experten

Vor allem begeistert das Spiel mit seiner Grafik: Die Gebirgsketten von Koronis Rift ziehen perspektivisch perfekt an Ihrem Cockpit vorbei, und an Bord Ihres Mutterschiffs wacht ein toll animierter Roboter über die Beutestücke. Auch bei den anderen drei Lucasfilm-Spielen sind spannende Handlung und schnelle Grafik Trumpf. »Ballblazer« ist eine Art Fußballspiel des 4. Jahrtausends, in dem die Bälle mit einem Magnetfeld eingefangen werden. Rescue on Fractalus« ist ähnlich wie »Koronis Rift« eine Planeten-Expedition, bei der diesmal abgeschossene Raumpiloten vor den finsteren Jaggis gerettet werden müssen. Der Vorspann dieses Spiels gehört übrigens mit zum besten, was man je an animierter Grafik auf einem Heimcomputer gesehen hat.

#### Abenteuerspiele in Deutsch und Englisch

The Eidolon« schließlich ist eine Expedition durch ein labyrinthartiges Höhlensystem, in dem es von merkwürdigen Kreaturen nur so wimmelt.

Abenteuerlustige Atari-Fans sitzen keinesfalls auf dem trockenen. Selbst wenn Sie keine Diskettenstation haben, können Sie mit Ihrem Computer ein saftiges Adventure spielen. Das englische

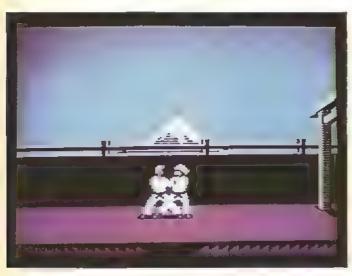
aber nur wenig halten. »Super Zaxxon«, die Fortsetzung zu »Zaxxon«, ist ein grafisch guter, aber spielerisch ausgesprochen liebloser Krawumm-Aufguß. Das Jump-and-Run-Spiel »Conan« hat mit dem barbarischen Leinwandhelden herzlich wenig zu tun. Die Handlung wirkt etwas an den Haaren herbeigezogen, und wer eine auffallende Ähnlichkeit zwischen Arnold »Conan« Schwarzenegger und dem mickrigen Bildschirm-Männlein feststellt, muß schon eine reichlich rege Phantasie haben. Selbst wenn man solche Hüpfspiele mag, überkommt einen rasch das große Gähnen. Da greife man lieber zum guten alten »Bruce Lee«, der viel mehr Spielspaß bietet.

#### Evergreen-Tips für die Sammlung

Überhaupt gibt es dank der Kompatibilität zur 400/800-Serie einige hinter dem Apple II-Original nicht verstecken braucht.

8-Bit-Computer gehören sicher zu den mittelfristig auslaufenden Modellen auf dem Markt: Der 800XL wird zu Schleuderpreisen ausverkauft. und der neue 130XE verzeichnet nur mäßige Absatzzahlen. Angesichts dieser Situation ist auch kaum mit einem wesentlichen Aufschwung beim Spiele-Angebot zu rechnen. Auf der anderen Seite dürfte der Nachschub von heute auf morgen keinesfalls völlig einschlafen, da vor allem in Deutschland und den USA noch größere Stückzahlen auf dem Markt sind. Wer aber in den nächsten Jahren in einem üppigen Software-Angebot schweigen will, sollte langsam einen Umstieg einplanen. Doch das Angebot an guten Atari-Spielen ist so groß, daß man ein ganzes Weilchen braucht, um sich durchzuspielen.

Zu guter Letzt noch ein Titel, der uns erst kurz nach Redaktionsschluß



Schlag auf Schlag: »Karateka«



Schöne Grüße vom »Ballblazer«-Finale

Softwarehaus Level 9 ist auf Abenteuerspiele spezialisiert und bietet auch einige Programme für die Ataris an. Zirka 30 Mark kostet Sie der Spaß eines Grafik-Adventures mit guten, allerdings englischen Texten und vielen Bildern. Nur auf Floppy gibt es das hervorragende Einsteiger-Adventure >The Dallas Quest« (Top-Grafik und Top-59 Mark) und Spielwitz, die Gehirnzellen-Feger »Mask of the Sun« und »Serpent's Star« (je 79 Mark). Der Bestseller »Mask of the Sun« liegt auch in einer voll eingedeutschten Version vor und nennt sich hier »Das Geheimnis der Aztekenmaske«,

#### Neben Tops auch Flops

An dieser Stelle mal ein paar Warnungen vor Neuerscheinungen der letzten Monate, die viel Spielwitz versprechen,

betagte Klassiker, die auch heute noch spielenswert sind. Wer seinen Atan noch nicht allzulang hat und verzweifett nach neuen Spielen sucht, sollte sich mal nach Evergreens wie »M.U.L.E.«, »Archon« oder »Dimension X« umhören. Letzteres gehört zu den wenigen Atari-Programmen, die nicht für den C 64 umgesetzt wurden. Einfacher Grund: Das Schießspiel, das im wesentlichen von seiner schnellen Grafik lebt, wäre in der Commodore-Version viel zu langsam geworden. Und wer sich nach der guten alten Computer-Steinzeit sehnt, kann sich immer noch den Original-»Pac Man« zulegen, der vor Jahren Ataris Superhit für das Videospiel-System 2600 war. Es werden aber hier und da noch Erfolgstitel anderer Computer nachträglich umgesetzt. So gibt es jetzt auch eine Atari-Version von Broderbunds langsamem, aber stilvollem Sport-Actionspiel »Karateka«, die sich

erreichte und den wir deshalb leider ohne Bild vorstellen müssen: »Mercenary« von »Encounter«-Autor Paul Woakes. Das Spiel ist eine einfallsreiche

### Neuer Hit mit 3D-Grafik

Mischung aus Flugsimulator, Geschicklichkeit, Strategie und Action und ist zu recht zivilen Preisen (42 Mark Kassette, 49 Mark Diskette) erhältlich. Das Glanzstück von »Mercenary« sind die vielen 3D-Vektorgrafiken, die diesen komplexen Spielgenuß zu einem garantierten Renner für 1986 werden lassen.

(Heinrich Lenhardt)

Bezugsquelle: Compy-Shop, Gnaisensustr 29, 4330 Mülhem/Ruhr, Tel. (0208) 497189 (USA-Importe) Funtastic, Tannhiluserplatz 22, 8000 München 80, Tel. (089) 939894 Teledienst, Mainzer-Tor-Anlage 45, 6360 Friedberg, Tel. (06031) 91650



## Cartridge-Experimente leichtgemacht

nichts mit der bekannten Messe in Las Vegas zu tun. Hier handelt es sich um ein Cartridge-Experimentier-System zum Selberbasteln.

s gibt verschiedene Wege, den Atari-Computer mit Software zu versorgen. Programme oder Daten können einerseits manuell über die Tastatur in den Arbeitsspeicher des Computers gelangen. Das ist aber, wie das Einlesen von Daten mit dem Kassettenrecorder, ein recht langwieriger Vorgang, Mit einer Diskettenstation geht es schon einige Größenordnungen schneller.

Die maximale Geschwindigkeit ist bei Verwendung eines ROM-Cartridges zu erreichen, da hier dem Mikroprozessor ein direkter Datenzugriff erlaubt ist. Als weiterer Vorteil wäre noch das problemlose Austauschen der Cartridges zu nennen. Ebenso erfreulich ist, daß nach dem Ausschalten des Computers die Informationen in den ROMs erhalten bleiben

Beim Atari 800XL/130XE ist für den Anschluß solcher Cartridges ein sogenannter Cartridge-Slot vorgesehen. Er befindet sich bei Computern der XL-Serie auf der Oberseite des Gerätes, beim 130XE an der Rückseite. In der Anschlußbelegung unterscheiden sie sich jedoch nicht.

An Software auf Cartridge sind sowohl Spiele als auch Programmiersprachen erhältlich. Je nach Programm werden 8KByte oder 16KByte-Cartridges eingesetzt. Diese ROM-Cartridges fügen sich, wie aus Bild 1 ersichtlich ist, in den Speicheradreß-

raum des Computers ein.

Ein Cartridge mit einem eigenen Programm war vielleich schon hin und wieder der Wunsch eines Atari-Besitzers. Oder wäre es nicht auch erfreulich, die Software eines käuflich erworbenen Cartridges den eigenen Bedürfnissen anpassen beziehungsweise erweitern zu können? Dieser Wunsch als Vater des Gedankens war der Anstoß für die Entwicklung der in Bild 3 gezeigten Schaltung.

Das CES (Cartridge-Experimentier-System) befindet sich auf einer Platine im Europaformat (100 mm x 160 mm). Es wird über eine Flachbandleitung und einen Cartridge-Slot-Stecker an den normalen Modulschacht des Atari-Computers angeschlossen. Die Platine des CES weist ebenfalls ein Slot zur Aufnahme der üblichen Cartridges auf. Ein Wahlschalter (SW 1) ermöglicht die Entscheidung, mit dem Cartridge der Platine oder mit CES zu arbeiten. Wird kein Cartridge benötigt, so muß es nicht extra entfernt werden. Eine eigene Funktion erlaubt die Trennung von CES beziehungsweise Cartridge und Computersystem.

Mit dem CES lassen sich verschiedene Cartridge-Größen simulieren. Es sind sowohl 8-KByte- als auch 16-KByte-Nachbildungen realisierbar. Für die Datenspeicherung sind zwei statische 8-KByte-RAM-Bausteine verantwortlich, die jederzeit eine Änderung der Cartridge-Software erlauben.



So sieht die fertig aufgebaute Schaltung aus



Die Schreibfreigabe-Leitung (WE) der RAM-Bausteine liegt normalerweise auf High-Pegel, so daß nur das Lesen der Module gestattet ist. Zur Programmierung des CES kann man jedoch mit Hilfe des Schalters SW 2 auf RAM-Betriebsart umschalten (Stellung »RAM«). Das CES verhält sich dann wie eine ganz gewöhnliche RAM-Erweiterung. Die Schreibfreigabe kann aber auch softwaregesteuert erfolgen (SW 2 in Stellung »Softswitch«). Dazu jedoch später mehr.

Zur Einstellung der Betriebsart des CES (Größe, Lage im Speicheradreßraum etc.) dienen zwei DIP-Schalter SA1 bis SA8 und SB1 bis SB4.

Zunächst einiges zu den am Cartridge-Slot des Atari angebrachten Anschlußleitungen. Da wäre einerseits der vollständige Datenbus, die Adreßleitungen A0 bis A12 sowie einige Steuerleitungen und eine 5-Volt-Spannungs versorgung.

Mit dem 13-Bit-Adreßbus läßt sich ein Speicheradreßraum von 8 KByte ansprechen. Über die Meldeleitungen M4 und M5 (Aktiv High) wird dem Computer mitgeteilt, in welchem Speicherbereich des Systems sich ein Cartridge befindet. Die Speicherverwaltungslogik im Atari weiß dann, an welchen Stellen sie das interne RAM ausblenden und das Cartridge einfügen soll. Dabei bedeutet:

M4 = High - 8 KByte Cartridge im Bereich zwischen \$8000 bis \$9FFF

M5 = High - 8 KByte Cartridge im Bereich zwischen \$A000 bis \$BFFF

Liegen beide Meldeleitungen auf High, so entspricht das einem 16K-Cartridge im Adreßbereich von \$8000 bis \$BEFE

Die Steuerleitungen S4 und S5 sind Low-Aktiv und kommen vom Computer. Sie dienen dem Chip-Select (Baustein-Auswahl) der ROMs im Cartridge. S4 wird Low, wenn ein Zugriff auf den Speicherbereich zwischen \$8000 bis \$9FFF erfolgt. Bei einem Zugriff auf den Bereich von \$A000 bis \$8FFF wird S5 Low.

Diese Steuer- und Meldeleitungen werden über den Wahlschalter SW 1 entweder mit dem Cartridge-Slot auf der Platine des CES oder mit den entsprechenden Anschlußpunkten der CES-Logik verbunden; abhängig davon, ob ein normales Cartridge im Slot steckt oder das CES betrieben werden soll. Befindet sich SW 1 in Mittelstellung, laufen die Steuer- und Meldeleitungen leer. Der Computer registriert dann, daß kein Cartridge angeschlossen ist.

Eine weitere Steuerleitung am Cartridge-Slot des Atari ist die

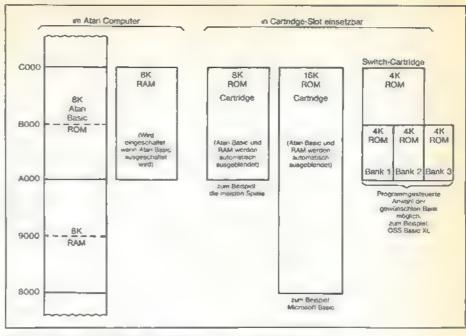


Bild 1. So fügen sich die verschiedenen Cartridge-Arten in den SpeicheradreBraum der Atari-Computer ein

Schreibfreigabe-Leitung WE. Sie steht immer dann auf Low, wenn in eine Speicherstelle geschrieben wird. Die Schreibfreigabe-Signale werden nun beim CES über eine Tor-Schaltung an die WE-Anschlüsse der RAMs geleitet. Damit ist es möglich, ein Beschreiben derjenigen RAMs, die zum CES gehören, zu verhindern (ROM-Betrieb).

Die Anschlüsse des Cartridge-Slot am Atari reichen schon aus, um ein normales Cartridge mit dem CES zu simulieren. Mit den beiden noch verbleibenden Anschlüssen PHI2 und CCTL lassen sich noch einige Zusatzfunktionen realisieren.

Am Anschluß PHI2 steht der Systemtakt des Atari zur Verfügung, von dem alle anderen Speichersteuersignale abgeleitet werden (zum Beispiel Schreibfreigabe). Besondere Bedeutung kommt jedoch dem Anschluß CCTL zu. Diese Leitung wird aktiv Low, wenn ein Zugriff auf den Speicheradreßraum \$D500 bis \$D5FF erfolgt.

Im Atari ist dieser Speicherbereich weder durch das RAM noch vom ROM belegt. Die CCTL-Leitung wird nun beim CES in Verbindung mit den untersten 6 Adreabusleitungen und dem Systemtakt PHI2 zur Software-Steuerung des CES benutzt. Mit Hilfe von IC 3 (6 Bit Dual-Flipflop) können die Zustände der Adreßleitungen AO bis A5 bei einem Zuariff auf den Speicherbereich \$D500 bis \$D5FF gelegt werden. Diese gespeicherten Zustände werden nun zur Steuerung des CES benutzt. Ein Beispiel soll dies verdeutlichen:

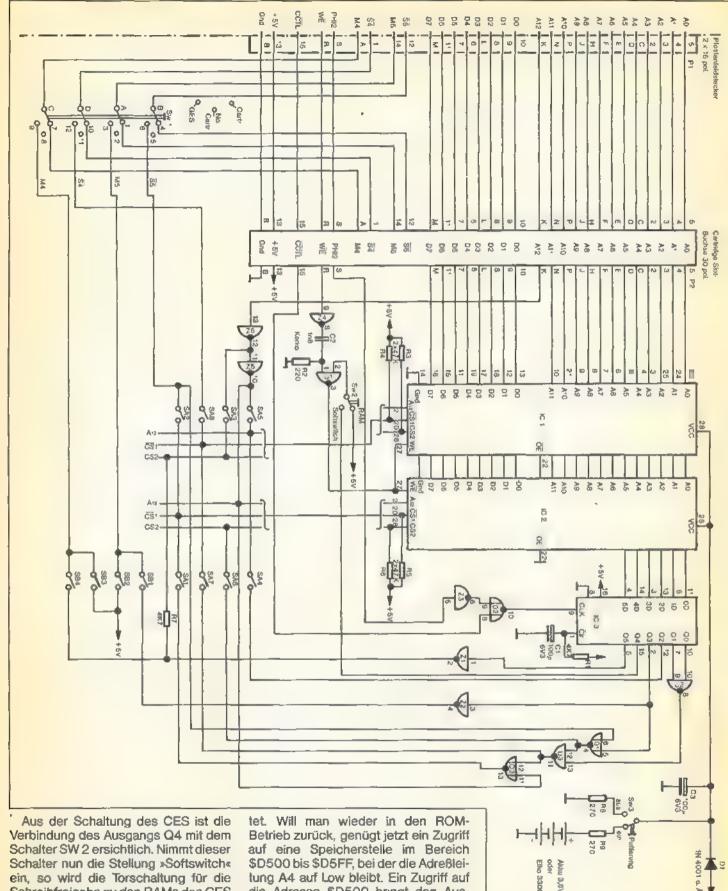
Soll ein Zugriff auf die Speicherstelle \$D510 erfolgen, so sind die unteren 6 Adreßleitungen zum Zeitpunkt des Zugnffs (Lesen oder Schreiben!) folgendermaßen gesetzt;

A0 bis A3 = Low A4 = High A5 = Low

Dieser Zustand wird nun in IC 3 gespeichert und liegt auch so lange an dessen Ausgängen Q0 bis Q5, bis eine andere Adresse im Bereich zwischen \$D500 bis \$D5FF angesprochen wird. So läßt sich dann ein neuer Zustand speichem. In unserem Beispiel liegt also an Ausgang Q4 ein High Pegel an.

Unterseit	Cartridge- Unterseite Ober					
S4	1	Α	M4			
A3	2	В	GND			
A2	3	С	A4			
A1	4	D	A5			
AO	5	Е	A6			
04	6	F	A7			
D5	7	Н	A8			
D2	8	J	A9			
D1	9	K	A12			
DO	10	L	D3			
D6	11	M	D7			
S5	12	N	A11			
+5V	13	Р	A10			
M5	14	R	WĒ			
CCTL	15	S	PHI2			
(*) Zeigt bei	(*) Zeigt bei der XL-Serie zur Tastatur					

Bild 2. Die Anschlußbelegung des Atari Cartridge-Slot



Aus der Schaltung des CES ist die Verbindung des Ausgangs Q4 mit dem Schalter SW 2 ersichtlich. Nimmt dieser Schalter nun die Stellung »Softswitch« ein, so wird die Torschaltung für die Schreibfreigabe zu den RAMs des CES über den High-Pegel des Ausgang Q4 von IC 3 geöffnet und somit über die Adreßleitung A4 gesteuert! In unserem Beispiel wird also, sobald sich SW 2 in Stellung »Softswitch« befindet, das CES in RAM-Betriebsart geschal-

tet. Will man wieder in den ROM-Betrieb zurück, genügt jetzt ein Zugriff auf eine Speicherstelle im Bereich \$D500 bis \$D5FF, bei der die Adreßleitung A4 auf Low bleibt. Ein Zugriff auf die Adresse \$D500 bringt den Ausgang Q4 von IC 3 auf Low-Pegel und sperrt damit wieder die Torschaltung. Das RC-Glied R1 bis C1 am CIr-Eingang von IC 3 sorgt dafür, daß nach dem Einschalten des Computers und des CES alle Ausgänge auf Low gesetzt sind.

Bild 3. Schaltplan zu CES (Cartridge-Experimentier-System)

reise
HM6264LP-1S (oder
ähnlich)
74LS174
74LS04
74LS00
74LS02
Drehschafter, drei Stel-
lungen mit vier
Schaltebenen
Schiebeschalter 1xUM
DIP-Schalter, 8xEm
DIP-Schalter, 4xEin
4,7 KOhm
220 Ohm
47 KOhm
270 Ohm
100 μ, 6V3 (Elko)
1n8 (Keramik-Elko)
weise ein 3.6-Volt-Akku-
rad-Elko

#### Bild 4. Bauteileliste zum Cartridge-Experimentier-System

Im CES werden für die Datenspeicherung statische CMOS-RAMs des Typs 6264LP-15 verwendet, die im nicht selektierten Zustand, also im Standby-Betrieb, eine sehr geringe Stromaufnahme haben. Deshalb ist in der Schaltung eine Pufferung der Spannungsversorgung für die RAM-Bausteine vorgesehen. So bleiben auch nach Abschalten der Computerspannungsversorgung die Daten im CES für längere Zeit erhalten. Man muß nicht unbedingt einen NC-Akku zur Pufferung einsetzen. Ein Elko mrt 3300 Mikro-Farad im CES erfüllt für immerhin ganze 11 Stunden denselben Zweck. Soviel zur Arbeitsweise der Schaltung.

Wichtig: Soli die Einstellung der DIP-Schalter des CES verändert werden, so müssen zuvor Computer und Pufferung des CES ausgeschaltet werden!

Nun einige Erläuterungen zu den Betriebsarten, wie sie in Tabelle 1 aufgeführt sind.

#### 8-KByte-Cartridges im Bereich zwischen \$8000 und \$9FFF

In dieser Betriebsart steht das CES direkt im Anschluß an den Adreßraum des eingebauten Basic zur Verfügung. Man kann dort zum Beispiel sehr einfach oft benötigte Hilfsprogramme unterbringen, Ist SW 2 in Stellung »RAM«, so kann man das CES von Basic aus wie ein RAM betreiben. In Stellung »Softswitch« erfolgt eine softwaremä-Bige Umschaltung zwischen RAM- und ROM-Betrieb. Wird die Speicherstelle 54544 dezimal (zum Beispiel mit »POKE 54544,1«) angesprochen, so wird der RAM-Betrieb eingeschaltet. Mit »POKE 54528.1 befindet man sich wieder im ROM-Betrieb.

Г	DIP-Schalter SA DIP-Schalter S8				Betriebsart des CES										
1		2	3	4	5	6	7	8	] 1	1	2	3	4		
0	,	0	0	0	0	0	0	1	(	, (	0	1	0		8K-Cartridge - Bereich \$8000 - \$9FFF
0	}	0	0	0	0	0	0	1			0	0	1		- Bereich \$8000 - \$9FFF (softwaregesteuert) (ausblendbar)
0	'	1	0	0	0	0	0	0			1	0	0		- Bereich \$A000 - \$BFFF 16K-Cartridge
0	)	1	0	0	0	0	0	1	(	)  ·	1	1	0		- Bereich \$8000 \$BFFF
1		0	1	1	0	1	1	0	1		0	0	0		16K-Switch-Cartridge - Bereich \$A000 - \$BFFF (mit Bank-Switching)
1	1 = Schalter geschlossen 0 = Schalter geöffnet														

	Charter gesernose	Self D = Gorialter
1 000		
1010		ANSCAR
	; Ermoeglicht d	the Outel name and
1848	; Daten aus des	im Slot befind-
		dge biw. das Kopie-
1960	o nen von Daten	aus dem RAM in das
1878	; Cantridge-Exp	erimentier-System.
1666		
1100	: SECROT = Anda	ngsadr. Bew Endadr.
1110	; SRCTOP des	Quellspercherber.
1129	: DSTBOT = Anfa	ngsadr, des Ziel-
1138	; spei	cherbereichs
1140		
1150		
1170		; Zwischenspeicher
1188	1	a management of the same
1199	SRCBOT = #A000	
	SRCTOP = SBFFF	
	05TBOT - \$6888	
1220	CONSOL = SD01F	- CONCO - Yeston
	MMIEN = #D40E	: CONSOL-Tasten : NM1-ENABLE-Ctrl.
1258		7
1260	RAPICAR = \$0518	:Adr. f. Schneib
1270		;frexgabe (je nach
1266		;Betriebsart des
1298	ROMCAR = \$2500	(CES)
1318		:50erre (je nach
1328	1	;Betriebsart des
1330	i	(CES)
1348		
1350		
	SWSTRT	
1388		(Alle Interrupts
1390	STA TEMP	isperren'
1480		
1410		
1428		
1448	NTSTRT	sharton him din
1450	CHP 46	;Warten bis die ;'START'-Taste
1460	BNE HTSTRT	
1478		
	EMMRT	
1498		;Schreibfreigabe ;fuer CES
1510		, tuel oca
1520	1	
1538	TRANS	
1548		
1558		ÜP
1568 1579		¡Kopiere Datem von
	NACH :	:SRCBOT bis SRCTOP
1598		anach DSTBOT 44.
1600	CPY VON+2	
1618		
1629 1638		
	BER RETURN INEVON	
1658		
1668		
1670	INC VON+2	
	INCTO	
1690		
1708		
1720		
1738		
1748	3	
1.750	DISMRT	
1768		;CES auf ROM-Be-
1776		;trieb schalten.
1799	RETURN	
1999		; Interrupts frei-
1619		;geben.
1620	RTS	
1839		
1846		;Autostartfile
1868	ASTRT ; .WORD SWSTR	erzeugen I
1879		

#### ▲ Tabelle 1. Die Betriebsarten des CES

◆ Listing 1. Quelitext zum Lesen des Inhalts eines Cartridges (MAC/65)

#### 8-KByte-Cartridge im Bereich zwischen \$8000 bis \$9FFF; softwaregesteuert und ausblendbar

Diese Betriebsart erlaubt es, das CES per Programm außer Betrieb zu setzen und das darunterliegende RAM des Atari zu aktivieren. So erhält man im Adreßraum \$8000 bis \$9FFF zwei 8 KByte große Speicherbereiche. Einmal das RAM des Computers, dessen Inhalt nach Ausschalten des Atari verlorengeht; zum anderen das CES, dessen Inhalt durch die Pufferung mittels Akku oder Elko auch nach Abschalten des Computers erhalten bleibt.

Folgende Zustände sind zum Beispiel von Basic aus steuerbar:

»POKE 54528,1« - CES als ROM aktiviert (=Einschaltzustand)

»POKE 54544,1« - CES als RAM akti-

»POKE 54560,1« – RAM des Computers aktiviert.

#### 8-KByte-Cartridge im Bereich zwischen \$A000 bis \$BFFF

Dies ist wohl die am häufigsten anzutreffende Cartridge-Version für Atari-Computer. Die meisten Cartridge-Spiele liegen in diesem Speicheradreßbereich. Aber auch das Atari-Cartridge-Basic, wie es früher für die Atari 400 und 800 Computer geliefert wurde, befindet sich in diesem Bereich.

Vom eingebauten Basic der XL- und XE-Serie ist das CES in dieser Betriebsart nicht mehr zu kontrollieren. Das eingebaute Basic liegt nämlich ebenfalls im gleichen Adreßraum und ist deshalb bei eingeschaltetem CES desaktiviert.

Über die Maschinensprache ist jedoch ein Zugriff auf den Inhalt des CES möglich. Bei Listing 1 handelt es sich um ein solches Umschaltprogramm. Mit einem Assembler, wie zum Beispiel MAC/65, kann man Maschinenspracheprogramme für jeden beliebigen Adreßraum schreiben, assemblieren und dann auf Disk speichern. Ein solches Programmfile wird dann



Listing 2. Dieses Assembler-Programm dient zum Umschalten zwischen dem CES und einem eingesteckten ROM-Modul

beim Laden vom DOS automatisch in den dafür vorgesehenen Speicherraum geschrieben.

Es muß also im Prinzip folgenden Aufbau haben, damit es in den Speicher des CES geladen werden kann:

- Lade einen Wert, beispielsweise 0, nach \$D510 (RAM-Betriebsart des CES).
- Lade die Daten f
  ür den Adre
  ßraum
   \$A000 bis \$BFFF in das CES.
- Lade einen weiteren Wert nach \$D500 (ROM-Betriebsart des CES).

Die Vorgehensweise für das Laden von Daten in das CES geschieht folgendermaßen:

- Den Computer und die Pufferung des CES ausschalten.
- DIP-Schalter für die entsprechende Betriebsart einstellen.
- Schalter SW 1 auf Stellung »CES«.
- Computersystem booten und Pufferung des CES einschalten.
- Das File mit oben beschriebenem Aufbau vom DOS aus laden.

Die über DOS geladenen Daten befinden sich nun im CES.

#### 16-KByte-Cartridge im Adreßbereich zwischen \$8000 bis \$BFFF.

Hierbei handelt es sich um zwei 8-KByte-Cartridges, die einen zusammenhängenden Adreßraum des Computers überdecken. Eine Kontrolle von Basic aus ist auch hier dadurch verhindert, daß das Basic von der oberen

Hälfte des 16-KByte-Bereichs überlagert wird und somit nicht zugänglich ist. Es bleibt nur noch der Weg über die Maschinensprachebene.

#### 16-KByte-Switch-Cartridge im Adreßbereich zwischen \$A000 bis \$BFFF

Wie aus Bild 1 ersichtlich handelt es sich hier um eine etwas komplexere Betnebsart. Das CES bietet die volle Speicherkapazität von 16 KBvte, belegt aber nur 8 KByte Adreßraum im Computer. Dies macht das sogenannte Bank-Switching möglich. Der Bereich von \$B000 bis \$BFFF ist normalerweise immer eingeblendet. Ein dort befindliches Maschinenprogramm kann nun durch Zugriff auf bestimmte Speicherstellen zwischen \$D500 bis \$D5FF einen der drei Speicherbänke im Bereich \$A000 bis \$AFFF aktivieren und zu einem dort befindlichen Unterprogramm verzweigen. Nach Abarbeitung des Unterprogramms erfolgt der Rücksprung in das aufrufende Programm im Bereich von \$B000 bis \$BFFF. Jetzt kann noch eine andere Bank eingeblendet werden, um eventueil ein dort vorliegendes Unterprogramm auszuführen.

Mit Hilfe dieser Technik passen 16-KByte-Module in nur 8-KByte-Speicher! Die einzelnen Speicherbänke werden durch Zugriff auf folgende Adressen aktiviert:

Adresse	Bank
\$D5X0	Bank 1 (untere 4 KByte in IC 1)
\$D5X3	Bank 2 (untere 4 KByte in IC 2)
\$D5X4	Bank 3 (obere 4 KByte in IC 1)
x-0	> ROM-Betriebsart
X=1	> RAM-Betriebsart

Nach diesem Prinzip arbeiten übrigens auch die Super-Cartridges von OSS. Will man Software in dieser Betriebsart ins Cartridge laden, so ist schon während des Ladevorgangs von Disk die Hardware des CES auf die entsprechenden Speicherstellen im Bereich ab \$D5XX einzustellen. Ein Programmfile sollte also folgenden Aufbau haben:

- 1. Lade einen Wert, beispielsweise 0, nach \$D510 (entspricht RAM-Betrieb des CES und Bank 1 ist eingeschaltet).
- 2. Lade die Daten für Bank 1 nach \$A000 bis \$AFFF und unmittelbar danach den Bereich von \$B000 bis \$BEFF
- Lade einen Wert nach \$D513 (entspricht dem RAM-Betrieb des CES. Bank 2 ist eingeschaltet).
- 4. Lade die Daten f
  ür Bank 2 nach \$A000 bis \$AFFF.
- Lade einen Wert nach \$D514 (entspricht RAM-Betrieb des CES, Bank 3 ist eingeschaltet).
- 6. Lade die Daten für Bank 3 nach \$A000 bis \$AFFF.

 Lade einen Wert nach \$D500 (entspricht ROM-Betrieb des CES. Bank 1 ist eingeschaltet).

Das Programm befindet sich nun im CES und kann aufgerufen werden.

Soviel an Hinweisen zum Gebrauch des CES (für Anwender), die eigene Cartridge-Programme schreiben wollen. Es würde den Rahmen sprengen, hier noch auf Details einzugehen, wie zum Beispiel das Betriebssystem die verschiedenen Cartridgearten behandelt, wann ein Cartridge-Selbststart erfolgt und wie das Betriebssystem registriert, ob ein Cartridgewechsel erfolgt ist. Mehr dazu kann man der entsprechenden Literatur entnehmen (zum Beispiel DeReAtari, Atari 600 XL/800XL inter von Data Becker) oder durch Experimentieren herausfinden.

CES erlaubt also, wie schon zu Beginn bemerkt, den Inhalt von normalen Cartridges herauszufinden. Im Prinzip geht das folgendermaßen vor sich:

 Im Cartridge-Slot der CES-Platine befindet sich das zu untersuchende Cartridge.

 Das CES wird auf die gleiche Betriebsart wie das Cartridge eingestellt

 Mittels eines kleinen Maschinenprogramms wird der Inhalt des Cartridges ins RAM kopiert

 Anschließend wird auf CES-Betrieb umgeschaltet, und die im RAM vorliegenden Daten werden, ebenfalls mit einem kleinen Maschinenprogramm, ins CES verlagert.

Der Inhalt des CES kann nun mit einem Monitorprogramm untersucht und disassembliert werden. Schaltet man das CES auf RAM-Betneb, so lassen sich auch Änderungen vornehmen. Ebenso besteht die Möglichkeit, den Inhalt des CES auf Diskette zu speichern und später wieder zu laden.

Das in Listing 2 abgedruckte Programm erledigt den Datentransfer zwischen Cartridge und RAM, zum Beispiel für ein 8-KByte-Cartridge im Speicherbereich zwischen \$A000 bls \$BFFF. Eine Änderung dieses Bereiches läßt sich recht einfach realisieren. Das Programm kopiert die Daten aus Speicherbereich dem zwischen SRCBOT und SRCTOP (im Listing beispielsweise \$A000 bis \$BFFF) in den Speicherbereich ab DSTBOT (im Listing beispielsweise \$6000 bis \$7FFF). Dieses Programm wird assembliert und unter dem Namen »READOUT.COM« auf Diskette gespeichert.

Dann wird das Programm noch einmal assembliert, jedoch mit den Adressen \$6000 für SRCBOT, \$7FFF für SRCTOP und \$A000 für DSTBOT und mit dem Namen »WRITECES.COM« auch noch auf Diskette gespeichert. Und so wird's gemacht:

- Computersystem und CES-Pufferung ausschalten.
- Zu untersuchendes Cartridge in den Slot des CES stecken.
- Betriebsart des CES entsprechend dem zu untersuchenden Cartridge einstellen.
- Schalter SW 1 in Stellung »CES«.
- Schalter SW 2 in Stellung »Softswitch«.
- Computersystem und Pufferung des CES einschalten.
- Bootprozeß abwarten, bis sich das DOS meldet.
- Das Programm »READOUT.COM« vom DOS aufrufen.
- Schalter SW 1 auf Stellung »Cart.« schalten.
- START-Taste drücken und abwarten, bis sich das DOS wieder meldet
- Programm »WRITECES.COM« vom DOS aufrufen.
- 12. Schalter SW 1 auf Stellung »CES« schalten.
- START-Taste drücken und warten, bis sich das DOS wieder meldet.

Im CES befinden sich nun die gleichen Daten wie in dem zu untersuchenden Cartridge. Jetzt läßt sich der Inhalt des CES beispielsweise mit einem Monitorprogramm wie Bug/65 untersuchen und disassemblieren.

Soviel zum Umgang mit dem CES. Wie der Name schon sagt, handelt es sich um ein Experimentiersystem. Beschäftigt man sich intensiv mit dem System, so wird man bestimmt noch andere, hier nicht aufgeführte Betriebsarten entdecken.

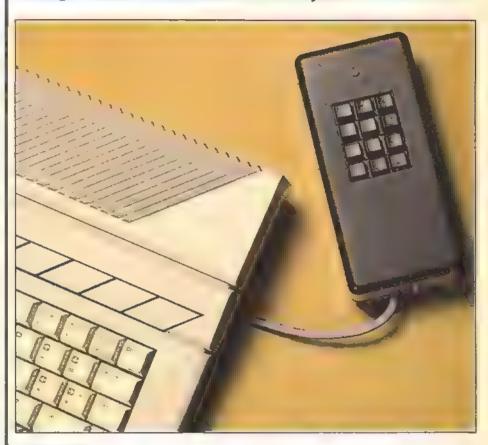
Zum Schluß noch einiges zum hardwaremäßigen Aufbau der CES-Platine. Wie bereits erwähnt, erfolgte die Realisierung der Schaltung mit einer 100 mm
x 160 mm großen, einseitigen Lochrasterplatine mit Lötpunktraster. Die Verdrahtung ist in Fädeltechnik ausgeführt
und deshalb schnell durchführbar. Zu
einem späteren Zeitpunkt könnte man
sich natürlich auch eine Platine entwerfen. Auch Änderungen lassen sich
schnell und einfach vomehmen.
Schließlich handelt es sich um ein
Experimentiersystem.

Die nötigen Anschlüsse zum Cartridge-Slot des Atari-Computers sind auf Pfostenfeldstecker (2 x 15 Pole) geführt. Die Verbindung zum Cartridge-Slot des Atari übernimmt ein 30adriges Flachbandkabel. An der Seite zur CES-Platine ist eine 34polige Pfostenfeldbuchse in Schneidklemmtechnik angepreßt. An der Cartridge-Slot-Seite zum Atari ist das Flachbandkabel an eine doppelseitige Streifenleiterplatine angelötet und mit Zwei-Komponenten-Klebstoff fixiert.

(H. D. Jankowski/wb)

## PIA - die dritte Schnittstelle

Mit sehr einfachen und preiswerten Mitteln läßt sich sogar eine selbstgebastelte Zusatztastatur an Ihren Joystick-Port anschließen.



▲ Die Zusatztastatur läßt sich problemlos in einem handlichen Gehäuse unterbringen.

PROGRAMM-STECKBRIEF			
Programmname	PIA-Steuerung		
Programmtyp	Utility		
Programmiersprache	Basic und Maschinensprache		
Programmlänge	insgesamt 6272 Byte		
für Computer	800 XL/130 XE		
zusätzliche Hardware	laut Bauanlertung		
Eingabehilfe	Prüfsummer		
Bemerkung	In Verbindung mit der Zusatzschaltung läßt sich eine Tastatur oder ein Schalt- interface steuern		
Lesersenvice	Diskette (PIA1.BAS/PIA2.BAS)		

as ganze Geheimnis der Joystick-Schnittstelle ist die sogenannte PIA. Es handelt sich hier um einen Standardperipheriebaustein des Typs 6520. Dieses Bauteil verfügt unter anderem über zwei 8 Bit breite Ports, die sich sowohl für die Einals auch für die Ausgabe von Daten nutzen lassen. Wie man diese Ports verwendet, ist dabei softwaremäßig völlig frei programmierbar. Leider macht sich

hier jedoch ein Nachteil der Atari XLund XE-Generation deutlich bemerkbar. Während die alten Modelle noch über vier Joystickports verfügten, sparte man bei den neueren Modellen zwei Ports ein. Durch diese Einschränkung konnte' der zweite Port der PIA natürlich nicht mehr nach außen geführt werden. Besitzt man also ein neueres Gerät, so kann man nur noch mit einem der beiden PIA-Ports arbeiten und ver-



#### HANDWARE BASTELE

fügt statt über insgesamt 16 Datenbits nur mehr über 8. Selbst ein hardwaremäßiger Eingriff, bei dem man den fehlenden Port nach außen führen würde, ist nicht durchführbar. Dieser Port übernimmt nämlich nun andere Aufgaben, vor allem in der Zusammenarbeit mit der MMU (Memory Management Unit oder Speicherverwaltungseinheit).

#### Dem Atari geht ein Licht auf

Im Normalzustand ist die PIA auf Eingabe programmiert. So ist es einsichtig, daß man bei der Auslenkung des Joysticks eigentlich nichts anderes tut, als einzelne Bits des Ports zu setzen oder zu löschen. Aus dieser Tatsache erklären sich auch die auf den ersten Blick etwas kuriosen Werte, die man mit Hilfe des STICK-Befehls erhält. Lenkt man den Joystick nicht aus, erhält man beispielsweise den Wert 15. Dies bedeutet, daß alle 4 für einen Joystick zuständigen Bits gesetzt sind (1111 binär ergibt 15 dezimal). Ist der Joystick nach



Auf der abgebildeten Testplatine wurden die Schaltungen von Bild 2 und Bild 3 auf einer Lochrasterplatine realisiert

oben gelenkt, so wird das niedrigste Bit gelöscht und man erhält den Wert 14 (1110 binär ergibt 14 dezimal). Welche Joystickbewegungen die einzelnen Bits beeinflussen, können Sie aus Tabelle 1 entnehmen.

Will man nun die Daten der PIA abrufen, erweist sich die STICK-Funktion nicht als besonders sinnvoll. Zum einen erfaßt man pro Befehl nur 4 Bit, zum anderen lassen sich mit diesem Befehl keine Daten an die PIA senden. Deshalb bedient man sich besser direkt der hierfür vorgesehenen Speicherstellen. Jedem der beiden PIA-Ports sind zwei Speicherstellen zugeordnet. Einmal das PIA-Register, das die zu sendenden oder zu empfangenden Daten enthält. Dann das Datenrichtungsregister, das festlegt, welche Bits des PIA-Registers für die Eingabe beziehungswelse Ausgabe verwendet werden (die dafür relevanten Adressen finden Sie in Tabelle 2).

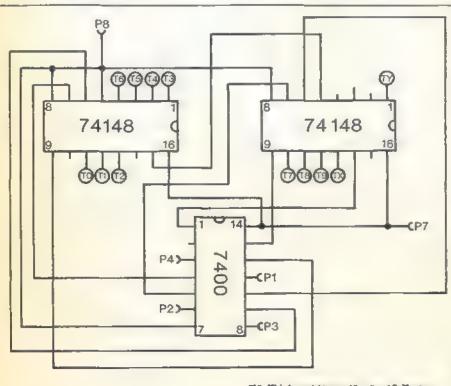


Tabelle 1. Für die Modelle 400/800 gelten dieselben Werte für Joystickanschluß 3 und 4

Port A	54016
Port B	54017
Port A DRR	54018
Port B DRR	54019
Joystick 1	632
Joystick 2	633
Joystick 3	634
Joystick 4	635

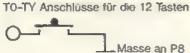
Tabelle 2. Der Zusatz »DRR« steht für »Datenrichtungsregister«. Die Adressen für Joystick 3 und 4 betreffen nur die alten Modelle.

■ Bild 1. Schaltplan für eine Zusatztastatur



- 12 Taster, Typ Digitast
- 2 IC 74LS148
- 1 IC 74LS00
- 1 Stecker 9pol.
- Kabel 6pol.
- 1 Gehäuse
- 1 Lochrasterplatine

Bauteileliste für eine Zusatztastatur



P1-P8 Pinnummer am Stecker um Computer

_	44.41	~	· · · · ·	21,001
1	2	3	3 4	5_
To	0	0	0	0/
1	0	0	0	0/
_	6	7	8	9

Pin 5, 6 und 9 werden nicht belegt

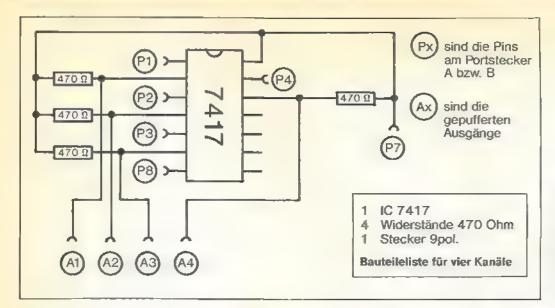


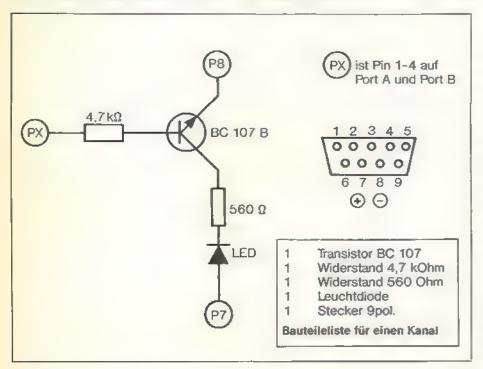
Bild 2. Schaltplan zum Ansteuern von vier Kanälen. Die abgedruckte Schaltung wird in zweifacher Ausführung benötigt, falls Sie alle acht Kanäle ansprechen möchten.

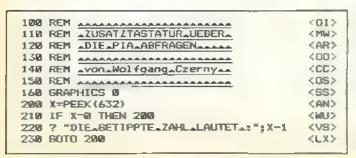
Die Programmierung der PIA geht folgendermaßen vor sich. Zuerst teilt man dem entsprechenden Datenrichtungsregister mit, daß die nächste Anweisung an den Port die Datenflußrichtung festlegen soll. Dies geschieht, indem man den Wert 48 in das Register POKEt. Der Wert für die gewünschte Flußrichtung ergibt sich dadurch, daß man für jedes Datenbit, das auf Ausgabe programmiert werden soll, eine

logische Eins und für jede Datenlertung zur Eingabe eine logische Null eingibt. Will man also beispielsweise alle acht Leitungen zur Ausgabe benutzen, ergibt dies den Wert 255 (oder binar

#### Daten rein oder raus?

11111111). Für die Eingabe aller acht Leitungen ergäbe sich also der Wert





ter Ausgang für einen Kanal. Entsprechend werden acht Schaltungen benötigt, um alle Kanäle anzusprechen.

4 Listing 1.
Ein kurzes
Beispielsprogramm zur Tastaturabfrage

▲ Bild 3. Gepuffer-

Null (binar 00000000). Man kann also ohne weiteres auch beide Flußrichtungen gleichzeitig festlegen, indem man sich den entsprechenden Wert errechnet

Der nächste Befehl bezieht sich nun wieder auf das Datenrichtungsregister. Man stellt den Normalzustand durch POKEn des Wertes 52 her, Nun ist das Port-Register nur noch für die logischen Zustände der Ein- und Ausgabe verantwortlich. Belegt man diese Speicherstelle also beispielsweise mit dem Wert Null, so würden alle Ausgänge den logischen Wert Null annehmen. Die Hardware, die man benötigt, um tatsächlich externe Vorgänge zu steuern oder zu erfassen, kann sich relativ einfach gestalten. Die Logikpegel der PIA entsprechen nämlich dem TTL-Standard. Dies bedeutet, daß die auszugebenden Informationen ohne Probleme durch recht preiswerte integrierte Bausteine weiterzuverarbeiten sind. Bei der Eingabe genügt es sogar im einfachsten Fall (wie es beispielsweise bei den Joysticks praktiziert wird), einen Eingang auf +5 Volt zu legen, um ein logisches Signal »Eins« zu erhalten.

#### Die Zusatztastatur

Als ein Beispiel, wie man externe Daten aufnehmen kann, wird hier eine Zusatztastatur beschrieben. Diese Tastatur besteht aus 12 Tasten, könnte aber auf bis zu 15 erweitert werden. Neben den Tasten besteht diese Schaltung im wesentlichen aus nur drei ICs. Einem 7400, er enthält vier NAND-Gatter und zwei ICs des Typs 74148. Dies sind sogenannte »8 zu 3 Prioritätsencoder« (der Schaltplan für die Zusatztastatur ist Bild 1 zu entnehmen). Der Vorteil dieser Schaltung liegt darin, daß ein einziger Joystickport ausreicht. Es werden nur vier Datenleitungen be-

nötigt, um die maximal 16 verschiedenen Zustände zu melden. Die Versorgungsspannung für die ICs wird dem Joystick-Port entnommen, so daß man ohne zusätzliches Netzteil auskommt. Listing 1 zeigt, wie man die Tastatur abfragen kann. Ist keine Taste gedrückt, erhält man den Wert Null, ansonsten einen Wert zwischen 1 und 11.

Auch die Datenausgabe ist hardwaremäßig einfach zu realisieren. Die Schaltung in Bild 2 zeigt, wie man mit nur zwei ICs, acht Widerständen und acht Leuchtdioden bereits die logischen Zustände aller Datenleitungen sichtbar machen kann. Da sich die Gesamtschaltung aus zwei identischen Teilen zusammensetzt, sieht man in Bild 2 nur den Aufbau für vier Kanäle. Die beiden ICs des Typs 7417 übernehmen hierbei die Aufgabe der Pufferung. Sie sind notwendig, um die PIA vor Überlastung zu schützen. An die Ausgänge der Pufferbausteine kann man nun wie in unserem Beispiel, Leuchtdioden, aber auch Optokoppler oder TTL-kompatible Kleinrelais anschließen.

#### Daten sichtbar gemacht

Bis zu diesem Punkt kommt man auch mit dem Schaltinterface mit der Stromversorgung der Joystickports aus. Es ist somit kein zusätzliches Netzteil erforderlich. Begnügt man sich mit der Steuerung von Leuchtdioden, so kann man auch die Schaltung in Bild 3 verwenden. Auch hier gilt: Die abgedruckte Schaltung in Bild 3 bezieht sich wieder auf nur einen Kanal.

Pro Datenleitung benötigt man zwei Widerstände, einen Transistor und eine Leuchtdiode. Auch hier wird ein Pufferungseffekt erreicht, da der Strom der Leuchtdiode nicht mehr über die PIA, sondern über den Transistor läuft.

Das etwas umfangreichere Listing 2 zeigt, wie man diese Schaltungen ansteuern kann. Neben einem programmierbaren Lauflicht ist auch eine Zeitschaltuhr realisierbar, die eine der Datenleitungen zu einem vorgewählten Zeitpunkt ein- oder ausschaltet. Um das Lauflicht schnell genug zu machen, ist eine Maschinenroutine in das Listing aufgenommen worden. Sie verschiebt den Registerinhalt des Ports A jeweils um 1 Bit nach links oder rechts, abhängig von der Einsprungstelle. Alles weitere kann man dem Listing selbst problemlos entnehmen.

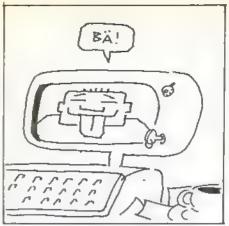
(Wolfgang Czerny/wb)

Listing 2. Mit diesem Programm kann bie Schaltung von Bild 2 als Schaltung von Bild 2 als Schaltunhr eingesetzt werden. Weiterhin ist noch eine Lauflicht-Funktion eingehaut

110 REM	<50> <50> <017 <017 <017 <000 <047 <000 <047 <047 <047 <047 <04
520 RESTORE 11000 530 FOR IM2 FO 14: RCAD DATE 540 POSITION 0,1:7 DATE 550 REXT 11 700 OPEN 01,4,0, "K:":SET 01,K:CLOSE 01 710 IF K:40 OF K:53 THEN 700 720 DN X-48 EDSUB 1000,2000,9000,5000,13 000 800 GOID 500	CNID CGED CUED CUED CUED CNID
999 RCH	(GI) (SY) (CT)
R3 (CTL R) (CTL R3 (CT	(X1) (VY) (GE) (FH) (00)
**)** 1130 FOR 1=17 TO 21:READ DATE 1140 POSITION 2,1:? DATE 1150 MCX7 I 1160 POSITION 33,10:? "C" 1170 PDME 764,235:R=0:CP=33 1175 FOR 1=0 TO 7:FL(1)=0:NEXT I 1200 TC=PEEX(764):SC=PEEX(5327?) 1210 IF 10=6 THEN POSITION CP,10:? "":C PRCP-4:4:4:150TO 1300 1220 IF TD=7 THEN POSITION CP,10:? "":C	CLA) CBO) CBO) CBO) CIP) CJT) CMC)
PHOP-44:F=F 1:GUTD 1300 1230 IF YC=28 OR TC=33 THEM FL (F)=(33=TC):GUTD 1400 1250 IF SC=5 THEN 1500 1260 IF SC=5 THEN 500 1270 IF SC=5 THEN 500 1270 IF SC=5 THEN FDIT=1:80TH 1000 1280 GUTU 1200 1300 IF FCO THEN F=0:CP=3 1340 POSITION CP:TOEN FCO:TC=3 1340 POSITION CP:TOEN FCSTION CP:TOEN FCO:TC=1400 IF PL (F)=0 THEN POSITION CP:TOEN TC=1400 IF FL (F)=1 THEN POSITION CP:TOEN TC=1410 IF FL (F)=1 THEN POSITION CP:TOEN TC=1400 IF FL (F)=1400 IF	(10) (50) (6N) (100) (100) (FE) (AU) (CE) (1L) (PD)
1420 PORE 764,255:GUTO 1200 1500 PODITION 2,23:7 "LM.INKSROTATION_/A PRECHISROTATION!" 1510 OPEN 61,4,6,"K:" GET MI,K:CLOSE W1 1520 IF K<76 THEN SHIFT=1536 1530 IF K<76 THEN SHIFT=1536 1540 IF K<92 THEN SHIFT=1555 1640 REM SIMMER:DEZ UMMANDLUNG 1700 NERT=0:FD 11-0 TO 71:FG (f):MERT=ME	CTE> CEN> CYY> CDE> CBA> CAS> CXD>
1700 MCRY-00:FDR 1-D YB 71X-FL ([]1MCRY-MC RT-2-19X-MCXT 1:FDVE P1,MCRY 1708 RCH SYART LAUFLICHT 1710 SC-PEEK (53279) 1720 IF SC-5 THEN GDTY-1:80Y0 1600 1730 IF SC-5 THEN GDTO 500 1730 GDUB 3000 X=USR (SHIFY) 1760 GDTO 1710 1999 RCH	CHZ) CJAJ CXL) CMR) CGZ) CTR) CRNO CENO CENO CWB)
2010 FER YHITE DIFTER 1=0 TO 255:PDKE P 1,1 GOSUB GOODINEXT TINEXT Y 2020 RETURN 2909 REM 5010 OF CHRISTON 5010 OF CHRISTON 5020 POSITION 2,717 "MELCHEN_KAMAL_SCHAL	<007>
5030 IF KAMALCI OR KANAL 98 THEN 5010 5035 POSITION 2,91? "MOLLEM_91E_CIN00C	(BA)
RAGUSCHULTON' SO4D POSITION 2,1117 "ElM=LANGHDA";:IMP UT EM 5050 IR EINKO OR EIN>1 THEN 5040 5090 IR EINHO YHEN POKE P1,223 5100 IR EINHO THEN POKE P1,233 5110 PIGITION 2,1512" "BERENASIEADIEAMOME	CMT> CTO> CAD> CAD>
3110 POGITION 2,13:7 "BEBEN_SIE_DIE_MOME NTAME_UNGTEIT-LIN" 5120 POSITION 2,14:7 "(STD,MIN,SEC)_";;; NPUT STD_MIN,SEC	(UF)
\$150 IF \$TD:23 OR \$TD<0 OR MIN\59 OR MIN (0 OR SEC\59 OR SECCO THEN \$120 \$140 UP1=\BTD=3600+MIN+60+SEC\950 \$150 UP2=\MT(UP1/256/256)	<2C>
\$155 UP1=UP1=UP2=256=256 \$160 UP3=INT (UP1/256):UP1=UP1=UP3=256	CFG>
3153 UP1-UP1-UP2-236-236 5160 UP3-INT (UP1/256): UP1-UP1-UP3-256 5163 POKE 16, UP2: POKE 19, UP3: POKE 20, UP1 5170 POSITION 2,16:7 "GEBRUSIE.DIE-SCHA LTZEIT-EIN" 5180 POSITION 2,17:7 "(STD, PIN, SEC) um: 1	CFC>

\$200 ? DARF(125) 5210 POSITION 11,61? TSCHALTUHR.LAGUETT	<6b>
5210 POSITION 11,4:7 "SCHALTUHRALAEUFT" 5220 POSITION 11,5:7 "************************************	(GP)
5230 IF FIN=0 THEN FINS-"AUS" 5240 IF FIN=1 THEN FINS-"EIN"	(FY)
3250 POSITION 1,827 "KANAL," EXAMAL & ".HTR D.UM."; STD1:":":MINN::":":SEC1:".":EINN:"	
SESCHAL TEY"	(DD)
5300 U1=PEEK(18):U2=PEEK(19):U3=PEEK(20) 5310 U1=INT (U1=254=254+U2=254+U3)/50)	<bh></bh>
5320 U2=1NT(U1/3600):U1=U1-U2=3600 5330 U3=1NT(U1/60):U1=U1-U3+60	< RQ>
5340 U2=U2 24+(U2)=24) 5350 POSITION 19,12:7 U2:":"1U3:":";U1	<ka></ka>
\$350 POSITION (0,12)? (2;":"1U3;":";U1 \$360 POSITION (0,12)? (2;":";U3;":",U) \$380 IF STD1COUZ OR MINICOUS OR SECICOU	<lb></lb>
THEN 9300	(HF) (DH)
5390 (F EIN-1 THEN POKE F1,2" (KANAL-1) 5400 (F EIN-0 THEN POKE P1,0 5900 RETURN	(GZ)
7999 REM	<pc></pc>
8000 REM VERZOEGERUNDSSCHLEIFE 8010 FOR DELAY-1 TO 50:NEXT DELAY 8020 RETURN	<ue></ue>
8999 REM	CPG>
9000 REH LOESCHEN LEDS PORT 1/2 9010 POKE P2,48	<dq></dq>
9020 POKE P1,255 9030 POKE P2,52	<6U>
9030 POKE P2,52 9040 POKE P1,0 9050 RETURN	<ut></ut>
10000 REP Maschinen File Jaden	<pe></pe>
10010 RESTORE 10100 10020 FOR 4=1536 TO 1579: READ D: POKE A, D	<cu></cu>
10030 REM LINKS-1536 / RECHTS-1555	<0X>
The state of the s	(UH)
13100 DATA_104,24,173,0,211,10,176,3,76, 19,4,141,43,4,149,1,76,33	<tu>&gt;</tu>
10110 DATA.6,104,24,173,0,211,74,176,3,7 6,39,6,141,43,6,169,170,24	CPT>
10120 DATALIO9,43,6,141,0,211,96,0 11000 RDM DATASLHRUPTHENUE	<00)>
11010 DATA_HAUPTHEND	<1145 <2145
11030 DATA 11040 DATA	<1,1>
11040 DATA 11050 DAYA.1LAUFLICHT 11060 DATA	<468>
11070 DATA_ZBINGER_LAUF	KHE>
11090 DATA_SALLE_LOSSCHEN 11100 DATA	<bb></bb>
11110 DATA_4SCHALTURE 11120 DATA	<2Y>
11130 DATA_5EMDE	(ZV)
11140 DATA 11150 DATA	<ha></ha>
11150 DATA 12000 REH DATAS_LAUFLICHY-MENUE 12010 DATA_CCTL Q3 CCTL R3 CCTL R3 (CTL R) C CTL W3 CCTL R3 CCTL R3 CCTL R3 CCTL W3 CCTL R3	<bv></bv>
SCELL MARCHE MARCHE MARCHE MARCHE	
RECOIL WESCHIL RESCOTE RESCOTE RESCOTE WESC TERESCOTE RESCOTE RESCOTE RESCOTE RE	
(CTL R) (CTL W) (CTL R) (CTL R	<04>
T =>===(SHIFT =>===(SHIFT =>===================================	<me></me>
T => (SHIFT =) (SHIFT => (SHIFT => (SHIFT =>	
[2040 DA7A_(SHIFT =) AAA_(SHIFT =) AAA_(SHIF	<100 >
7 =>(SMIFT =>(SMIFT =)(SMIFT =>	<hy></hy>
12050 DATA_(CYL_R)(CYL_	
) (CTL R) (CTL R) (CTL 5) (CTL R) (CTL R) (CTL	
RECOTL SECOTL RECOTL RECOTL RECOTL SECOTL SECOTL RECOTL	
A DALLA CATA AMILIET -> AMILIET -> AMILIET	<08>
12060 DATA_(SMIFT =)(SMIFT =)(SMIFT =)	<0色>
T -1 400/TCT -1 400/TCT -1 400/TCT -1	<0A>
T -1 400/TCT -1 400/TCT -1 400/TCT -1	
T -1 400/TCT -1 400/TCT -1 400/TCT -1	
T =>(SMIFT =>(SMIFT =>(SMIFT =>(SMIFT =>(SMIFT =>(SMIFT =>(SMIFT =>(SMIFT =>(SMIFT =>) 12070 DAYA(CTL. 2) (CTL. R)	
7 = 3(SMIFT = 3(SMIFT = 7)(SMIFT = 8)(SMIFT = 3(SMIFT = 3(SMIFT = 3(SMIFT = 3(SMIFT = 3)) 12070 SAYA(CTL 2) (CTL R) (CTL	<n@></n@>
7 = 3(SMIFT = 3(SMIFT = 7)(SMIFT = 8)(SMIFT = 3(SMIFT = 3(SMIFT = 3(SMIFT = 3(SMIFT = 3)) 12070 SAYA(CTL 2) (CTL R) (CTL	<n@></n@>
7 = 3(SMIFT = 3(SMIFT = 7)(SMIFT = 8)(SMIFT = 3(SMIFT = 3(SMIFT = 3(SMIFT = 3(SMIFT = 3)) 12070 SAYA(CTL 2) (CTL R) (CTL	<nb></nb>
T = 1(SMIFT = 2)(SMIFT = 2)(SMIFT = 3)(SMIFT = 3)	<n@></n@>
7 = 3(SMIFT = 3)(SMIFT = 3)	<nb></nb>
7 = 3(SMIFT = 3)(SMIFT = 3)	<nb></nb>
7 = 3(SMIFT = 3(SMIFT = 3) = 4.6MIFT = 3 = 3(SMIFT = 3(SMIFT = 3) = 12070 SMIFT = 3 12070 SMIF.(SMIFT = 3) = 12070 SMIFT = 3 2 CTL X) CCTL R) CCTL CCTL R) CCTL R	<nb></nb>
7 = 3(SMIFT e)(SMIFT e)	<nb></nb>
7 = 3(SMIFT e)(SMIFT e)	<nb></nb>
7 = 3(SMIFT e)(SMIFT e)	<nb></nb>
7 = 3(SMIFT = 3(SMIFT = 3) = 4.5MIFT = 3) = 12070 BAYA.(COL. 2) (CYL. R) (CY	<nb></nb>
7 = 1(SMIFT = )(SMIFT = )	<nb> <zn> <zno <bho="" <cn<="" <cno="" td=""></zno></zn></nb>
7 = 1(SMIFT = )(SMIFT = )	<nb></nb>
7 = 3(SMIFT = 3)(SMIFT = 3)	<nb> <zn> <zno <bho="" <cn<="" <cno="" td=""></zno></zn></nb>
7 = 3(SMIFT = 3)(SMIFT = 3)	<nb> <zn> <zno <bho="" <cn<="" <cno="" td=""></zno></zn></nb>
7 = 3(SMIFT = 3(SMIFT = 3) and (SMIFT	<nb> <zn> <zno <bno="" <n<="" <nbn="" td=""></zno></zn></nb>
7 = 1(SMIFT = )(SMIFT = )	<nb> <zw> <bh> <bh> <wo> <au></au></wo></bh></bh></zw></nb>







#### Mit einer einfachen und preiswerten Schaltung wird der Atari zur Steuerungseinheit für Elektromotoren. Damit läßt sich ein Robotarm dirigieren.

Industrieroboter übernehmen heute in vielen Fabriken die schmutzigen und schweren Arbeiten. Ein Atari wäre bei diesen komplizierten Produktionsabläufen natürlich überfordert. Doch für die Steuerung von kleinen Modellrobotern ist er hervorragend geeignet. Einen einfachen Robotarm zu bauen, ist gar nicht so schwierig. Die Konstruktionsbaukästen von Märklin, Trix oder von Fischertechnik, sind eine gute Grundlage. Zusätzlich benötigt man natürlich noch die entsprechenden Motoren und Getriebe. Einige Punkte sind bei der Planung zu beachten:

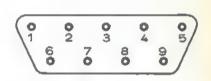
- Es sollten nur Gleichspannungsmotoren eingesetzt werden (6-12 V)
- 2. Alle Motore brauchen ein Getriebe, möglichst mit Schneckenrad-Antrieb
- Stabiler, mechanischer Aufbau mit großer Grundplatte
- 4. Beim Verlegen der Kabel Bewegungsachsen beachten
- Geeignete Stromquellen benutzen (Netzteile)

Entwerfen wir ein Konzept:

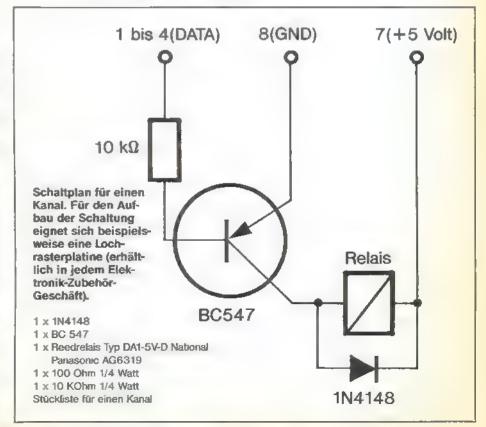
- Anzahl der Freiheitsgrade: Die Anzahl ist abhängig von der Funktion. Drei Freiheitsgrade sind in der Regel ausreichend, das heißt, drei bewegliche Achsen und drei Motore. Sie sind nötig zum Drehen des Grundkreises, heben und senken der senkrechten Achse, aus- und einfahren der waagerechten Achse.
- ☐ Größe des Verfahrweges: Der Verfahrweg ist individuell zu gestalten, er ist verantwortlich für den Aktionsradius des Armes.
- Positionierung eines Robotarmes: Um den Geldbeutel zu schonen, verwenden wir keine Schrittmotoren. Besitzer eines Atari 400/800 können

## Roboter im Griff

Pins	Beschreibung		
1 bis 4	DATA		
7	+5 Volt maximal 40 mA		
8	GND (Masse)		



Die Anschlußbelegung eines Joystickports



#### HARDWARE-BASTELET

sich freuen, weil hier Port 3 und 4 noch frei sind. XL-Besitzer dagegen müssen sich mit der Steuerung über den Timer zufriedengeben.

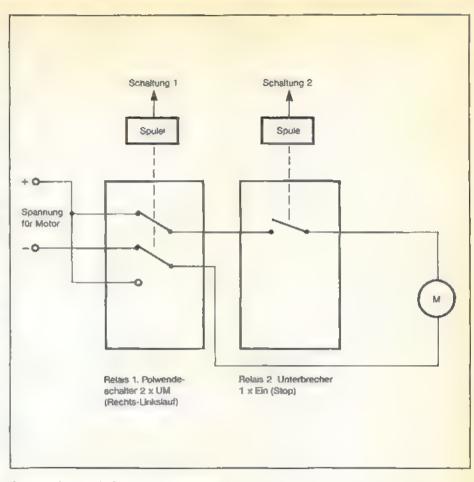
Der Atari-Computer ist natürlich nicht zum Steuern eines Robotarms ausgelegt. Wir müssen deshalb eine kleine Schaltung aufbauen (siehe Schaltplan). Damit sind wir in der Lage, den Arm von Basic aus zu steuern. Die Schaltung wird über Port 1 und 2 angeschlossen. Die Ports werden mit »POKE 54018.1« auf Datenausgabe umgeschaltet. Aber Vorsicht, nach Ausführung dieses POKE-Befehls darf kein Joystick mehr angeschlossen sein. Mit »POKE 54016,0-255« werden die Datenkanäle geschaltet. Jetzt steht uns eine Parallel-Schnittstelle mit 8 Bit zur Verfügung. Liegt ein Datensignal an, schaltet Transistor die computereigene 5-Volt-Spannung auf das Reedrelais, welches dann anzieht. Durch sinnvollen Einsatz der POKEs kann man allen Motoren gleichzeitig ohne Interrupt-Programmierung ansteuern. Beim Atari 400 oder 800 kann man zusätzlich noch Potentiometer an alle Achsen anbauen (Wert 2MOhm) und über Port 3 und 4 den Wert erfassen. Damit ist es moglich den Arm relativ genau zu posi-

Alle Atari-XL-Benutzer können nur über die Zeit positionieren und haben den Nachteil ungenauer Steuerung. Wird nämlich der Motor belastet (zum Beispiel durch ein Gewicht), läuft er langsamer und erreicht die gewünschte Position nicht mehr. Hardwarespezialisten können versuchen über die neue Schnittstelle zum Erfolg zu gelangen.

Das Basic-Programm ist für zwei Motoren ausgelegt und arbeitet mit einem Joystick in Port 4. Mit dem Joystick sind alle Bewegungen zu steuern. Pfiffige können das Programm so umschreiben, das sogar ein »Teach in« möglich wird. Das heißt, steuern des Armes zur gewünschten Position mit dem Joystick, übernehmen des Wertes in das Programm und danach automatischer Ablauf des Vorgangs. Als Greifer leistet ein umgebautes Relais gute Dienste. Damit kann man metallische Gegenstände greifen und transportieren. Ein preiswerter Elektromagnet funktioniert natürlich auch.

Zum Schluß noch einige Tips: Die Joystickports sollten pro Kanal nicht höher als mit 10 mA belastet werden. Die Spule der Relais muß also die Werte 5 V/500 Ohm haben. Die Positionierungspotis sollten untersetzt werden, damit der gewünschte Punkt nicht überfahren wird. Auch alle Getriebe sollten über Schnecken und Schnekkenräder untersetzt sein, damit eine Bremse überflüssig wird.

(Jürgen Warschinsky/wb)



Schematischer Aufbau der Schaltung

```
4 PRINT CHR$ (125)
5 FOKE 54018,1:X=54016:POSITION 10,8:? "
REWEGLING: "
10 A=STICK(3)
20 IF A=15 THEN POKE X,245: POSITION 10,1
Q:? "
         STOP
:0 IF A=14 THEN POKE X,249: POSITION 10,1
0:? "
          UP
40 IF A=13 THEN POKE X,253:POSITION 10,1
0:7 "
         NWOG
50 IF
      A=11 THEN POKE X,247: POSITION 10,1
0:? "
         LEFT
60 IF A=7 THEN POKE X,246:POSITION 10,10
: ?
        RIGHT
70 IF A=9 THEN POKE X,255: POSITION 10,10
:? "DOWN & LEFT "
80 IF A=5 THEN POKE X,254:POSITION 10,10
:? "DOWN & RIGHT"
90 IF A=10 THEN POKE X,251:POSITION 10,1
Ø:? "
      UP & LEFT "
100 IF A=6 THEN POKE X,250: POSITION 10,1
0:? " UP & RIGHT "
110 GOTO 10
```

Listing zur Ansteuerung des Selbstbau-Schaftinterfaces





Markt Cleebralk

uke/Puliko Der ATARI 520 ST 2. überarbeitete und erweiterte Auflage 1986, 198 Seiten

Dieses Buch enthält alle informationen, die für Intorossierte und für alle stolzen Besitzer eines gerade erworbenen ÄTÄ. 1520/260 ST wichtig sind. Die jetzt worliegende überarbeitete und erworkerte Auflage trägt den neuesten Entwicklungen bei Atan Rechnung, Unter ande rem wurden das inzwischen deutsch-sprachige Betriebssystem und einige geänderte Systemausstaltungsmerkmale berücksichtigt. Das Buch ist somit nach nur eine Rechnerbeschreibung mit hohem Informationswert, es leistet auch als Nachschlagewerk wertvolle Dienste Best-Nr MT 90229

ISBN 3-89090-229-4 DM 49,-/sFr. 45.10/65 382,20

#### Das Systemhandhuch ZILIN ATARI ST

2. Quartal 1986, ca. 300 Seiten

Zwei Themen bilden die Schwerpunkte des vorliegenden Buches: Die Struktur der 68000-CPU und der

ATAN 520 260 ST Die ausführliche Beschreibung der Ar-chitektur der 68000 Familie (68000, 68008, 68010, 58020) und ihrem Be-

68008, 68010, 58020) und ihrem Be-rehissatz wird erganzt durch einen Nach-schlagefeil mit zwei- bis dreizeiligen Be-spelsequenzen. Auf dieser theoreti-schen Basis wird die Programmenunge-bung des ATARI 520/260ST anhand-vieler Beispielprogramme dargestellt. Die Entwicklung dieser Programme lie-fert dem Leiser dieserbertin eine Riblicefert dem Leser gleichzerlig eine Biblio-thek mit Routinen zur Ansteuerung des Bildschimiteils, der Tonerzeugungs-schaftung und der Schrittstellen (MDI). V24, Tastatix, Maus). Besondere Auf-merksamkeit wird der Einbindung von Maschinensprachmodulen in das Betnebssystem und in höhere Programersprachen (z. 8. BASIC und C) gewid it. Die Besprechung eines 68000 Assembiers und einige gerätespezifi-sche Maschinensprachmodule runden

das Buch ab. Best. Nr. MT 90216 ISBN 3-89090-216-2 DM 52-/sFr 47.80/65 405.60



A. Steiner/G. Steiner GEM für den ATARI 520 ST 2. uberarbeitere und erweiterte Auflage 1986, 334 Seiten

Autlage 1986, 334 Setten

Die Benutzeroberflache des neuen
ATARI ST – GEM genannt – erhebt den
Anspruch, die Bedienung des Computers zum Kinderspiel zu machen. Den
noch. Wenn Se die bisher übliche kommandodrientierte Umgangsweise mit ihrem Computer pflegten so werden Sie
eine Enführung in die Bedienung von
Maus, Bildsymbalen und Fenster, wie sie
dieses Buch liefert, zu schältzen wissen.
Besonders interessant für den erfahre-Besonders interessent für den erfahre-nen Artwender sind die Kapittel über den internen Aufbau von GEM mit seinen Pull-Down-Menßs, Fenstern und Sym-

Best-Nr MT 90230 ISBN 3-89090-230-8 DM 52.-isFr, 47,80/6S 405,60

Yorki Zechnik Di C-Programm J. Purdum/T Leslie

Die C-Programmbibliothek 1. Quartal 1986, ca. 320 Seiten

Dieses Buch erspart dem C-Programmierer Stunden munseliger Kleinarbeit und hilft, effizientere Programme zu schreiben Esist in zwei Teile gegliedert. Der erste Teil zeigt, wie man zu universel-len Bibliotheksfunktionen kommt und gibt Tips, wie C noch wirkungsvoller ein-gesetztwerden karn. Der zweite Teil ent hålt eine Reihe ausführlich erklärter G-Funktionen als wertvolle Ergänzung Ih-rer Programmbibliothek. Dazu gehören unter anderem ein Terminalinstallabonsprogramm, mehrere Sortier-Algorithmen und ein Satz ISAM-Funktionen

ISBN 3-89090-133-6 DM 69,-/sFr. 63,50/6S 538,20

W Fasterirath

#### ATARI ST BASIC-Handbuch

1. Quartal 1986, ca. 250 Selten Suchen Sie eine Anleibung zur intensiven Ausnutzung der Fähigkeiten des ATARI 520/260 ST? Dehn ist dieses Buch genau das richtige für Sie! Sie erfahren alles über das BASIC-System des ATARI ST. Jeder Befeh wird mit Programmbenspelen ausführlich erfäutert. Den Schwerpunkt bildet eine Anleitung zur BASIC-Programmening des ATARI ST sowie zur Programmening von GEM-Funktionen. Best.-Nir IIIT 90205, ISBN 3-89090-205-7, DM 52,-IsFr.47,80/6S 405,60

#### AYARI ST LOGO

1. Quartal 1986, ca. 250 Seiten

Dieses Buch bietet eine gründliche Einführung in die Programmersprache LOGO und ihre Anwendung auf dem ATRI 520/260 ST Schon nach kuizer Zeit ist der Anfänger in der Lage, eigene LOGO-Programme zu sohreiben. Praktische Anwendungsmöglichkeiten wie z. B. die Dalenverweitung sind auch für den für-geschriftenen Programmerer von Interesse. Ein expenes Kapitel ist dem Bereichder künstlichen Intelligenz

gewomen Best-Nr. MT 90223, ISBN 3-89090-223-5, OM 49,-hifr.45,10/6S 405,60

P Rosenbeck

#### C-Programmerung unter TOS/ATARI ST

1. Quartal 1986, ca. 300 Seiten

Die Programmiersprache C hat sich bei professionellen Programmierem zu einem Renner entwickelt. Sie ermöglichtes, sehrnabe an der Maschine zu arbeiten und doch strukturientzu programmieren. Dieses Buch bieter eine Einführung in die Programmierung C speziell für dem ATAPI ST. Am Beespiel eines Diskettermoni-tors wird die Systemprogrammierung gründlich und umfassend erfautert. Außerdem erfahren Sie alles über den Einsatz von BIOS-Routinen und über das Software-Eingenering Best-Nr. MT 90226, ISBN 3-89090-226-X, DM 52,-/sFr. 47,80/6S 405,60

In Vorbereitung:

C-Programmerung unter GEM/ATARI ST

2. Quartal 1986, ca. 300 Seiten Best, Nr. NT 90263, ISBN 3-89090-203-0, DM 58,-/sFr. 53,40/6S 452,40



W. Hilf-A. Nausch M68000-Familie: Teil 1 1984, 568 Seiten

Informative Einführung in die Geschich te und die Entwicklungsphilosophie er-ner detaillierten Darstellung der Hardware sowie ausführliche Erläuterung der

komfortablen Adressierungsarten Best.-Nr PW 705 ISBN 3-921803-16-0 DM 79.-/sFr 72.80/6\$ 616.20

M68000-Familie: Teil 2 1985, 400 Seiten

Teil II des umfassenden Lehr- und Nachschlagewerkszum M68000 beschäftigt sichmit Anwendungen und weiteren Mit gliedem der M68000-Familie. Best.-Nr. PW 713

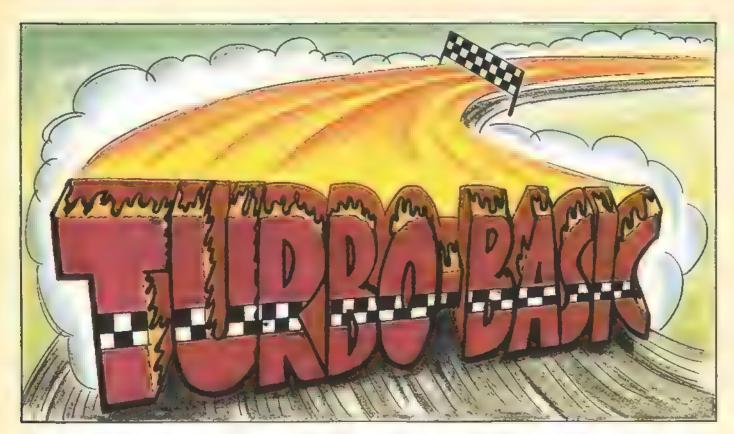
ISBN 3-921803-30-6 DM 69,-IsFr 63,50/6S 538,20

#### Figrid & Technik Eschburger erhalten Sie bei Ihrem Buchhändler

Bestellkarten bitte an Ihren Buchhändler oder an einen unserer Depot-Händler. Adressenverzeichnis am Ende des Heftes.

Bestellungen im Ausland bitte an unterstehende Adressen. Schweiz: Markt & Technik Vertriebs AG, Kollerstrasse 3, CN-6300 Zug, 27 042/415656 Österreich: Rudolf Lechoer & Sohn, izwerkstraße 10, A-1232 Wien, 🏖 02 22/67 75 26

Markt&Technik BUCHVERLAG



## Der große Turbo-Basic-Teil

Turbo-Basic XL ist extrem schneil, komfortabel und erlaubt Programmstrukturen wie Pascal. Der Turbo-Basic-Compiler, der sich nun zur Turbo-Serie hinzugesellt hat, steigert das Programmiervergnügen und die Geschwindigkeit noch einmal.

urbo-Basic XL werden manche Leser bereits kennen (der Interpreter wurde in Happy-Computer, Ausgabe 12/85 abgedruckt). Eventuell haben Sie auch schon erste Erfahrungen mit diesem leistungsfähigen Interpreter gesammelt. Dann werden Sie bestimmt nicht mehr allzuhäufig mit dem normalen Atari-Basic arbeiten und lieber auf das Turbo-Basic XL zurückgreifen. Aus diesem Grund veröffentlichen wir den Interpreter noch einmal in diesem Sonderheft. Wir sind nämlich der Meinung, daß sich Turbo-Basic XL unter möglichst vielen Atari-Fans verbreiten sollte.

Wenn Sie den Interpreter allerdings noch nicht kennen, schenken Sie den nächsten Seiten Ihre Aufmerksamkeit. Dort finden Sie eine ausführliche Erklärung zu allen Turbo-Basic-XL-spezifischen Befehlen. Hinzu kommen einige neue Fehlermeldungen und die Möglichkeit, verschiedene Disketten-Befehle direkt von Basic auszuführen. Las-

sen Sie sich aber nicht von dem umfangreichen Listing abschrecken. Denn haben Sie das Programm erst einmal abgetippt, werden Sie es nicht bereuen. Dessen sind wir ganz sicher.

Gleich im Anschluß an den Turbo-Basic-XL-Interpreter finden Sie einen weiteren Leckerbissen: Den Turbo-Basic-Compiler, ebenfalls zum Abtippen. Einen superschnellen Compiler, der selbst in Atari-Basic geschriebene Programme verarbeitet. Das Compilat zählt mit zu den schnellsten. Überzeugen Sie sich selbst von den Ergebnissen. Auf der Seite 32 sind die erzielten Rechenzeiten aufgelistet. Unser Compiler hat selbst kommerzielle Compiler um Längen geschlagen.

Damit sich die beiden »Wunderkinder« auch gleich bewähren können, sind einige Turbo-Basic-XL-Programme hier abgedruckt. Beachten Sie aber bitte: Diese Beispielprogramme laufen nicht unter Atari-Basic! Es kommen nämlich spezielle Turbo-Basic XL-Befehle zur Anwendung, die Atari-Basic nicht verarbeiten kann. So ist beispielsweise im Programm »Grafikspielereien« der »MOVE«-Befehl eingebaut, der ein sehr schnelles Verlagern des Bilschirmspeichers bewirkt. So wird ein Grafikbild mal auf den Kopf gestellt oder über die Mittelachse hinweg gespiegelt. Aber lassen Sie sich überraschen! Das Programm bietet viele wirkungsvolle Effekte, die sich ansonsten in

Atari-Basic nur durch spezielle Maschinensprach-Routinen realisieren lassen.

Oder werfen Sie einen Blick auf das Programm »Apfelmännchen«, Dieses Programm erzeugt die tollsten Grafiken. Ein farbiges Beispiel finden Sie im Inhaltsverzeichnis am Anfang des Heftes. Hier kommt ein weiterer Vorzug von Turbo-Basic XL voll zur Geltung, nämlich der, strukturiert zu programmieren. Es wurde vor allem der Befehl »EXEC Prozedur« eingesetzt. So lassen sich selbst Prozeduren unter Turbo-Basic Beim genaueren definieren. Studium des »Apfelmännchen«-Programms wird Ihnen sicherlich auffallen, daß in dem Programm kein einziger GOTO- oder GOSUB-Befehl auftaucht. Dies ist eine der wichtigsten Voraussetzungen der strukturierten Programmie-

Es spricht also einiges zugunsten unseres Interpreters und Compilers. Um noch mehr Atari-Besitzer in den Genuß des Turbo-Basic XL-Interpreters zu bringen, ist eine Umsetzung der beiden Programme auch für den Atari 260 ST/520 ST+ in Vorbereitung. Natürlich ist in dieser Version auch die Befehls-Kompatibilität zwischen dem 800XL und den ST-Computern gewährleistet. Allerdings benötigen die Programme noch einige Zeit bis zur Vollendung. Wir halten Sie aber auf dem laufenden.

(wb)



## So funktioniert der Turbo-Basic-Compiler

Soll ein Basic-Programm schneller werden, dann greift man einfach zum Basic-Compiler, Aber wie funktioniert ein solches Beschleunigungswunder?

anche Anwendungen erfordem mehr Tempo, als reine Basic-Programme zu bieten haben. Dann schreibt man zeitkritische Routinen entweder in Maschinensprache oder setzt einen Compiler ein, der ein Basic-Programm in Maschinensprache umwandelt. Letzteres ist naturlich viel einfacher und beguemer, da es lediglich den Umgang mit dem Compiler vorraussetzt. Außerdem benötigt die Compilierung nicht viel Zeit. Das Compilat kann übrigens nicht mehr von Basic aus bearbeitet werden. Wenden wir uns nun der Funktionsweise des Turbo-Basic-Compilers zu.

Wenn Sie ein Basic-Programm eintippen, zum Beispiel die Programmzeile \*100 ?3\*4«, faßt der Computer diese Zelle zunächst nur als eine Folge von Zeichen im ATASCII-Format auf. Unter dem Begriff ATASCII versteht man die spezielle Atari-Version des ASCII-Zeichensatzes (ASCII ist die Abkürzung für: American Standard Code for Information Interchange). Der Interpreter wandelt den eingegebenen Text in Token um. Gleichzeitig überprüft er noch die Syntax, also ob die Befehlsfolge auch zulässig ist. Diese Aufgabe muß meistens auch ein Compiler übernehmen. Bild 1 zeigt, wie die Programmzeile »100 ?3 \* 4 c in Token übersetzt aussieht.

Die Programmzeile wird dann anhand ihrer Nummer einsortiert. Durch diese Umwandlung in Token und gleichzeitiger Syntaxprüfung wird die Aufgabe des Compilers etwas einfacher. Speichert man ein Basic-Programm mit dem SAVE-Befehl, werden auch die Token gespeichert. Mrt LIST hingegen liegt es in der Textform vor. Der Turbo-Basic-Compiler kann nur die »SAVE-Ausführung« übersetzen. Ein Transfer der Text-Form ware prinzipiell auch möglich. allerdings würde der Compiler hierfür mehr Zeit benötigen. Schließlich muß er die Befehlsnamen kennen und die Variablennamen spelchern. Dies würde weiterhin die maximale Länge der zu compilierenden Programme vermindern. Oder es müßten Zwischenfiles auf Diskette angelegt werden, was bei den relativ langsamen Atari-Diskettenlaufwerken sehr zeitaufwendig ist.

#### Keine Variable ohne Nummer

Bevor Sie ein Basic- oder Turbo-Basic XL-Programm compilieren, sollten Sie sicherstellen, daß es fehlerfrei läuft. Zwar ist dann immer noch nicht gewährleistet, daß das Programm anschlie-Bend auf Anhieb ordnungsgemaß compiliert und ausgeführt werden kann. Zumindest entfallen aber Fehler, die zur Folge hätten, daß Sie Ihr Basic-Programm korrigieren müßten. um es anschließend nochmals zu compilleren. Auch muß die Struktur Ihren Basic-Programme stimmen. Dazu bietet sich der LIST-Befehl unter Turbo-Basic XL an, der bekanntlich FOR-NEXT-Schleifen und IF-ELSE-ENDIF-Abfragen einrückt.

Atari-Basic und Turbo-Basic XL-SAVE-Files bestehen aus einem Vorspann mit den Variablennamen und einigen anderen Informationen für den Interpreter. Diese werden deshalb vor dem Einlesen entweder überlesen (wie die Variablennamen) oder, in einer angepaßten Form, gespeichert. Dies betrifft in erster Linie die Variablentypen. Der Variablentyp kann aber nicht nur aus der Variablennummer, die im Programm gespeichert ist, definiert werden. Das Dollarzeichen (\$) beispielsweise, welches die Stringvariablen kennzeichnet, wird vom Interpreter nicht im eigentlichen Programm gespeichert.

Bei tangen Programmen mit vielen Variablen kann man diese Phase der Compilation deutlich erkennen. In der Info-Zeile (der dritten Bildzeile des Turbo-Basic-Compilers) steht dann nur das Wort Zeile ohne Zeilennummer. Sobald die erste Zeile übersetzt wird, erscheint auch die Anzeige der entsprechenden Zeilennummer.

Jetzt beginnt die eigentliche Compilierung. Sie setzt sich aus zwei Durchgängen zusammen, im ersten Durchgang (Paß 1) wird das Programm Zeile für Zeile und Befehl für Befehl eingelesen und in Maschinencode übersetzt. Im zweiten Durchgang (Paß 2) werden dann noch die offenen Sprungadressen eingesetzt. Dazu werden im Paß 1 GOTO-Sprünge im Code durch einen ungültigen Maschinenbefehl mit angehängter Zeilennummer gekennzeichnet. Im zweiten Durchgang schließlich ersetzen dann 6502-JMP-Befehle (Code \$4C) die ungültigen Maschinen-

Ähnliches gilt für Zahlen und Textkonstanten. Sie werden in einem Block hinter dem eigentlichen Basic-Programm gespeichert. Die endgültigen Adressen stehen dann erst nach der Übersetzung fest. Im Anschluß an den zweiten Compilier-Durchgang folgt dann noch eine Kontrolle, ob offene Schleifen vorliegen. Also ob ein NEXT zu einem FOR fehlt oder ein WEND zu einem WHILE etc. Trifft dies zu, erscheint eine Meldung auf dem Bildschirm, Dann folgt noch die Überprüfung nach fehlenden ENDIFs. Durch diese Aufteilung der Fehlerprüfung (die meisten Fehler werden bereits im ersten Durchgang festgestellt) werden die fehlerhaften Zeilennummern nicht komplett sortiert ausgegeben, sondern gegebenenfalls in vier einzelnen Gruppen.

#### Nach der Umwandlung in Token sieht die Programmzeile »100 ? 3°4« tolgendermaßen

OA 00 15 15 28 0E 40 03 00 00 00 00 24 0E 40 04 00 00 00 00 18

#### Dabei bedeuten:

0A 00

= Zeilennummer (100)

15

= Zeilenlänge 15

Abstand vom Zeilenanfang zum Ende der ersten

Programmzeile = Token für »?«

28 OF 40 03 00 00 00 00

= 2abl 3

24

= Token für »\*«

QE 40 04 00 00 00 00

= Zahi 4 = Token für Zeilenende

Bild 1. Eine Basic-Programmzeile, umgesetzt in Token



Die meiste Arbeit wird im ersten Durchgang geleistet, der Rest dauert nur wenige Sekunden. Das Einlesen des Programms erfolgt Zeile für Zeile. Zuerst zur Zeilennummer: Wenn die Zeilennummer größer als 32767 ist, dann ist Paß 1 beendet (der Interpreter verwendet die Zeilennummer 32768 übrigens als Endekennzeichen). Dann schließt der zweite Durchgang an. Ansonsten wird die Zeilennummer angezeigt und die Zeile Befehl für Befehl eingelesen und übersetzt.

Sollte der Compiler auf einen LISToder ENTER-Befehl stoßen, wird einfach eine Fehlermeldung mit Zeilennummer ausgegeben, und der nächste Befehl kommt an die Reihe.

Keinerlei Probleme bereiten REModer >--«-Zeilen. Diese Befehle werden einfach überlesen und ignoriert, da sie den Programmablauf nicht beeinflussen.

Befehle ohne Parameter (wie END, POP oder CLR) werden einfach zu einem entsprechenden Unterprogrammaufruf compiliert.

Bei einem DATA-Befehl wird der Text mit Zeilennummer und Länge in einer Tabelle im Anschluß an das Programm eingetragen. Sie nimmt noch folgende Informationen auf:

- eine weitere Tabelle mit konstanten Zahlen
- konstanten Strings
- die Anfangsadressen aller Programmzeilen
- den w\u00e4hrend der Compilation ben\u00f6tigten Stapelspeicher (der 6502-Stapel ist viel zu klein)

Diese Informationen werden gegebenenfalls auch im Speicher verschoben, wenn das zu compilierende Programm oder eine der Tabellen anwächst. Deshalb nimmt auch die Compilierungsgeschwindigkeit bei langen Programmen gegen Ende ab; vor allem wenn viele DATAs am Anfang des Programms stehen. Diese Verlangsamung betrifft lediglich die eigentliche Compilierung und nicht den Programmablauf.

Alle Befehle, die Parameter erfordern, sind wesentlich schwieriger zu übersetzen. Bild 2 zeigt nur einen Befehl mit einem einzigen Parameter.

Die Umwandlung des Befehls von Schritt 1 nach Schritt 2 hat der Interpreter schon erledigt. Der Compiler liest das Befehlstoken \$03 für COLOR und verzweigt in die entsprechende Compilationsroutine. Diese ruft wiederum die Routine für die Übersetzung eines Ausdrucks auf, die eine Integerzahl zurückliefern soll. Dann wird nur noch »STA \$C8« angehängt, und damit ist die Compilierung des Befehls schon erledigt. Beim Befehl GRAPHICS würde »JSR @Graphics« angehängt werden. Der Klammeraffe (@) kennzeichnet übrigens eine Routine in der Runtime Bibliothek.

Ein großer Fortschritt wurde bis jetzt noch nicht verzeichnet. Zwar sind die Befehle jetzt prinzipiell compiliert, das ändert aber leider nichts daran, daß mit dem entsprechenden Parameter, wie in unserem Beispiel die 1, noch nichts passiert ist.

#### Compilieren Zeile für Zeile

Die dafür benötigte Routine gliedert sich in mehrere Abschnitte:

- P-Code-Erzeugung
- P-Code-Optimierung
- Maschinencode-Erzeugung

P-Code (Pseudo-Code) ähnelt der Maschinensprache eines hypothetischen (gedachten) Prozessors. Manche Compiler erzeugen nur einen P-Code, der dann von einem kleinen Interpreter ausgeführt wird. Die Abarbeitungsgeschwindigkeit eines solchen Codes ist zwar schneller als ein herkömmliches Basic-Programm, allerdings von der Geschwindigkeit einer >richtigen« Maschinensprache noch weit entfernt. Aber platzsparender ist P-Code im Vergleich zu voll compiliertem Basic, Echte Maschinensprache ist jedoch von keinem Compiler zu übertreffen.

Bild 3 zeigt ein komplizierteres Beispiel. »03« entspricht wieder dem Token für COLOR, »46« steht für PEEK, »3A« für die offene Klammer, »2C« für die schließende Klammer, »35« steht für die Addition (+), »16« bedeutet Zeilenende oder auch einen Doppelpunkt, der Basic-Befehle in einer Zeile trennt. Der Wert »0E« kennzeichnet schließlich

noch eine Zahl in der internen Darstellung (entspricht sechs Byte).

Das Token »03« ist in unserem Beispiel bereits identifiziert. Es beginnt jetzt der Test, in welcher Reihenfolge die Operationen ablaufen müssen. Der Interpreter hat dabei fast die gleichen Tests durchzuführen, und zwar jedesmal, wenn er diese Zeile erreicht. Der Compiler übersetzt jede Zeile nur einmal bei der Compilierung. Dadurch laufen die Programme schneller ab.

Bei der Übersetzung in P-Code bedient sich der Compiler einer Tabelle, in der die Prioritäten aller Operatoren vermerkt sind. Übrigens handelt es sich hierbei fast um die gleiche Tabelle, die auch der Interpreter benutzt. Daraus ergibt sich dann folgender P-Code. In Bild 4 sind die einzelnen Werte lediglich in Kurzform dargestellt. Der Compiler betrachtet sie natürlich als Zahlen.

Jetzt könnte theoretisch schon die Routine aufgerufen werden, die den Maschinencode erzeugt. Allerdings würden dann im compilierten Programm die gleichen Berechnungen ablaufen, die der Interpreter normalerweise auch durchführt. Schließlich steht jetzt viel Zeit zur Optimierung des P-Codes zur Verfügung, da die Programmzeilen nur ein einziges Mal in Maschinencode umgesetzt werden.

Hier nochmals der P-Code:

Zahl 41 02 00 00 00 00

fp->intpeek int->fp push

Zahl 40 01 00 00 00 00

movepull add fp->int

im ersten Schritt faßt der Optimierer die Zahl »41 02 00 00 00 00 fp->int« zusammen, indem er den Wert In die Integer-Darstellung umwandelt. Daraus entsteht:

Izahl 200 peek int->fp push Zahl 40 01 00 00 00 00 movepull add fp->int

Anmerkung: Es gibt zwei Fp-Accus, \$D4 bis \$D9 und \$E0 bis \$E5. Integerzahlen stehen dabei im Prozessorregister A und Y. Dabei enthält Y das höherwertige und A das niederwertige Byte.

Der Optimierer faßt jetzt »Izahl 200 peek« zu »Ipeek 200« zusammen. Das erste Beispiel würde »LDA # < 200:LDY # > 200:JSR PEEK« als Code erzeugen und das zweite Beispiel »LDA 200:LDY # 0«. Jetzt haben wir also:

Ipeek 200 int->fp push Zahl 40 0100 00 00 00 movepull add fp->int

Jetzt wird »push Zahl ... movepull add« zu »add# ...« zusammengefaßt. Daraus ergibt sich:

1) COLOR 1

2) 03 0E 40 01 00 00 00 00

3) A9 01 85 C8

,so sieht's aus

,so speichert's der Interpreter

das soll daraus werden (LDA #1.STA \$C8)

;\$C8=200 ist die Speicherzelle, in der sowohl

Interpreter als auch Compiler die Zeichenfarbe

für Plot ablegen

Bild 2. So wird der Befehl »COLOR« compiliert

03 46 3A 0E 41 02 00 00 00 00 2C 35 40 01 00 00 00 00 16

Bild 3. So sieht der Basic-Befehl »COLOR PEEK(200)+1« in Token umgesetzt aus

Ipeek 200 int->fp add
# 40 01 00 00 00 00 fp->int

Als nächstes bemerkt der Optimierer die Folge »int->fp Add # ... fp->int« und versucht diese Befehlsfolge zu vereinfachen. So spart man sich später eine Umwandlung. Da die Zahl kleiner als 256 ist, gelingt dies auch.

Ipeek 200 Iadd# 1

Jetzt hat der Optimierer seine Arbeit geleistet. Anschließend wird richtiger Maschinencode erzeugt.

Aus »lpeek 200« entsteht »ŁDA
200:LDY #0«.

»ladd # 1« wird zu »LDX #1:JSR IADD« (dieser Unterprogrammaufruf ist nötig, da sonst auch ein Overflow stattfinden könnte; Integerwert größer als 65535!)

Das war auch schon alles. An den Befehl COLOR wird noch »STA \$C8« angehängt. Insgesamt ergibt sich also:

LDA 200:LDY # 0:LDX # 1:JSR IADD:STA \$C8

Ohne Optimierung würde wesentlich mehr Programmcode erzeugt werden, der noch dazu viel langsamer ist:

LDA # < Z200:LDY # > Z200:JSR

IDOYA:JSR FPI?:JSR PEEK:JSR IFP:JSR PUSH IDA # <Z1:LDY # > Z1:JSR MOVEPULL:JSR ADD:JSR FPI?:STA \$C8

Außerdem noch im Konstantenbereich:

Z200 .BYTE \$41,\$02,\$00,\$00,\$00, \$00 Z1 .BYTE \$40,\$01,\$00,\$00,\$00, \$00

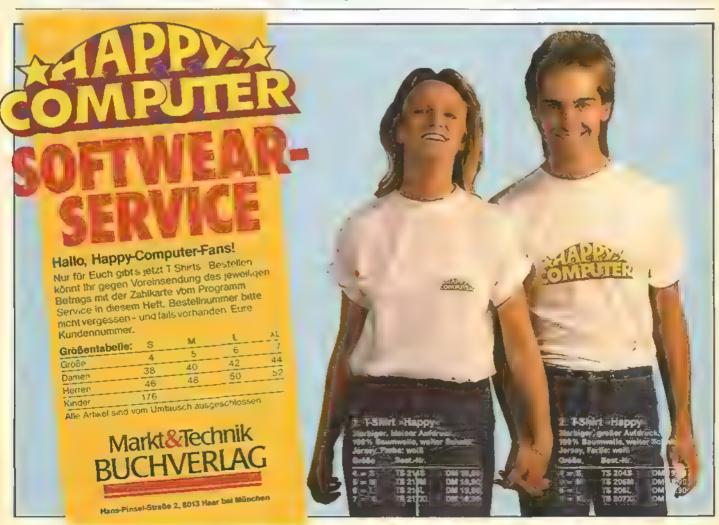
Durch Optimierung läßt sich also viel Programmcode sparen. Gegenüber einem echten Maschinencode sieht das Ergebnis allerdings immer noch recht umfangreich aus. In Maschinensprache würde nämlich »INC \$C8«, was nur 2 8yte benötigt, das gleiche bewirken. Hexadezimal \$C8 entspricht dabei dem Dezimalwert 200. In unserem Beispiel codiert der Compiler den Basicbefehl COLOR 1 jedoch optimal, nämlich »LDA #1:STA \$C8«.

#### **Optimal** compiliert

Allerdings kann der Compiler nur in den seltensten Fällen optimalen Code erzeugen. Schließlich erübrigt sich nur die Umwandlung von FP nach INT und umgekehrt und dies auch nur bei Verwendung von Konstanten. Bei Variablen ist diese Art der Optimierung nicht zulässig. In den meisten Fällen reicht schon die Einsparung einiger Bytes, weil zum Beispiel das Laden einer Variablen und die Umwandlung in einen statt zwei Integerwerte durch einen einzigen Unterprogrammaufruf erledigt wird.

Auch wird nicht alles, was möglich wäre, optimiert. Zum Beispiel berechnet sich »A=10/3« nicht im voraus, obwohl es durchaus sinnvoll ware. Aber solche Optimierungen kann der Programmierer gegebenenfalls selbst vorwegnehmen. Allerdings sind solche Fälle viel zu selten, als daß sich dies hier lohnen würde. Je länger nämlich der Compiler, desto kürzer die entsprechenden Basic-Programme. Es gibt jedoch einige Berechnungen, deren Optimierung sich besonders lohnt » 2« erhält. eine eigene, schnelle Routine. Im Compiler benötigt »A=B^2« die gleiche Zeit wie »A=B\*B«. Es existiert ebenfalls eine schnelle Exponential-Routine, die bereits im Mathematik-Teil der Runtime-Bibliothek vorhanden ist. So oder ähnlich werden alle Befehle mit Parametern behandelt. Sind mehrere Parameter vorhanden, werden diese naturlich zwischengespeichert.

Ein weiterer lohnenswerter Befehl ist



;die 200 Zahl 41 02 00 00 00 00 ;der PEEK-Befehl benötigt einen Integer als Adresse, fp->int peek int->fp deshalb muß die 200 zuerst umgewandelt werden ;Dann folgt der eigentliche PEEK-Befehl und dann die Umwandlung des Ergebnisses in eine Floating-Point-Zahl (fp). Diese "Umwandlungen muß der Interpreter auch sinner .durchführen. retten des Ergebnisses auf den Rechenstapel push Zahl 40 01 00 00 00 00 :Zahl 1 Movepult Schiebt die Zahl 1 aus dem FP-Accu 1 in den :FP-Accu 2 und holt das Ergebnis des PEEKs in den FP Accu 1 :die Addition Add dies wird von der Compilationsroutine für Color fp->int ;angehängt, es wird ja ein Integer benötigt

Bild 4. Der P-Code, der sich aus »COLOR PEEK(200)+1« ergibt

der POKE-Befehl. Beispielsweise wird 
POKE 16,64< durch >LDA #64:STA 
16, übersetzt.

Beim PRINT-Befehl sind noch die Kommata zu beachten. Auch INPUT, READ, GET und LOCATE beinhalten noch einige Probleme. Eine ausführliche Erläuterung würde den Rahmen dieses Artikels jedoch sprengen.

Gesondert zu behandeln sind alle Befehle, die den linearen Programmablauf beeinflussen, Der GOTO-Befehl läßt sich vergleichsweise einfach übersetzen. Zuerst wird die Routine aufgerufen, die einen Integer-Ausdruck übersetzt und optimiert (wie oben); allerdings generiert sie ihn noch nicht in den endgültigen Maschinencode, Zunächst wird nur der P-Code erzeugt. Dann wird anhand des P-Codes überprüft, ob eine Zahl folgt. Wenn ja, wird ein JMP-Befehl compiliert. Zunächst nur »07 Zeile«, woraus sich nach Paß 2 »4C Adresse« ergibt. Wenn nein, wird die Routine für die Erzeugung von Maschinencode aufgerufen und »JSR GOTOX« angehängt. Bei einem berechneten GOSUB wird beispielsweise entweder »JSR GOSUBX« oder vor dem »JMP-(\$07)«-Befehl der GOTO-Anweisung ein Unterprogrammaufruf compiliert. Dieser enthält dann, auf dem Basic-Stack, die um drei Werte erhöhte Rücksprungadresse des Befehls. Es wird aber nicht einfach ein JSR compiliert, da sonst die Anzahl der Unterprogrammebenen durch den zu kleinen 6502-Stack begrenzt wäre. Außerdem müssen die Parameter für die FOR-NEXT-Schleifen ebenfalls auf diesem Stapel abgelegt werden. Sie belegen jeweils 16 Byte, so daß sehr schnell ein Overflow des 6502-Stack drohen würde. Dies könnte katastrophale Folgen haben und die Kompatibilität zum Interpreter-Basic stark einschränken. Zumindest ist die untere Hälfte der Seite 1 frei, und es gibt Programme, die mehr als 10 geschachtelte Schleifen verwenden. Der Zeitverlust durch diesen simulierten Stack hält sich iedoch in Grenzen.

Einen ähnlichen Stack verwendet der

Compiler auch zur Übersetzung von DO-LOOP, REPEAT-UNTIL, WHILE-WEND und FOR-NEXT-Schleifen sowie des EXIT-Befehls.

Bei DO wird die aktuelle Adresse auf diesem Stapel abgelegt. Sie erhält außerdem noch die DO-Befehlskennung. Bei Übersetzung des zugehörigen LOOP wird die Kennung daraufhin überprüft, ob ein DO fehlt und dann ein JMP-Befehl auf die zuvor festgestellte Adresse compiliert. Sollte zwischen DO und LOOP ein EXIT gestanden haben, so hat dieser Befehl seine eigene Kennung auf dem Stack hinterlassen und JMP-Befehl mit »Dummv«-Adresse erzeugt, LOOP ersetzt diese Adresse durch die auf den eigenen JMP-Befehl folgende.

#### Geschwindigkeit ist Trumpf

Bei REPEAT ist der Vorgang ähnlich wie bei DO. Es unterscheidet sich lediglich die Kennung. Wenn UNTIL auf eine REPEAT-Anweisung folgt, wird zuerst der folgende Ausdruck übersetzt. Erst dann wird ein bedingter Sprung angehängt. Da der 6502 nur relativ kurze, bedingte Sprünge kennt, wird mit »BNE \*+5:JMP ad« gearbeitet. Natürlich muß auch UNTIL eventuelle EXIT-Befehle berücksichtigen.

WHILE-WEND entspricht dem eben Besprochenen, nur steht hier die Bedingung am Schleifenanfang.

Bei FOR-NEXT-Schleifen ist noch zu beachten, daß zusätzlich der Basic-Stack benutzt wird. Deshalb wird automatisch ein POP-Befehl in den EXIT-Befehl eingeschlossen.

Damit ist die Wirkungsweise des Compilers erklärt. Vielleicht ist Ihnen aber noch nicht ganz klar, wie der Interpreter und der Compiler die Reihenfolge der Berechnungen feststellen. Der Interpreter führt die Berechnungen aus, der Compiler erzeugt einen P-Code. Trotzdem funktionieren beide nach dem gleichen Prinzip. Zur Zwi-

schenspeicherung der Operatoren dient jeweils ein Stack (Stapel). Dazu ein einfaches Beispiel:

8\*Sin(A)+16

Der Compiler liest die Ziffer 8. Als Zahl wird sie sofort compiliert. (Der Interpreter würde sie auf dem Zahlenstapel ablegen.) Dann wird das Token für Multiplikation (\*) mit der Priorität 5 gelesen. Da der Stapel für Operatoren noch leer ist (Priorität 0), wird das Token dort abgelegt. Als nächstes folgt SIN mit der höchstmöglichen Priorität (9). Weil diese größer ist als die auf dem Stapel (5), wird auch SIN auf den Stapel gelegt. Dann folgt die geöffnete Klammer. Als Spezialfunktion ruft die Klammer dieselbe Routine noch einmal, allerdings rekursiv auf. Es wird also der Wert in Klammern entsprechend übersetzt oder berechnet. Dann kommt die Variable A an die Reihe. Variablen werden, genauso wie Zahlen, sofort compiliert. Dann schließt als Endekennzeichen die Klammer ab. Weiter geht es mit der ursprünglichen Übersetzung. Es folgt nun das Token für Addition (+), dessen Priorität (4) kleiner als dle auf dem Stapel ist. Deshalb wird jetzt der SIN-Befehl compiliert. Anschließend wird noch die vorliegende Priorität mit der des \*\* «-Tokens (5) verglichen, Da sie wieder kleiner ist, wird jetzt die Multiplikation compiliert.

Der leere Stack hat die Priorität 0, also wird das Token für Addition auf dem Stack abgelegt. Dann folgt die Zahl 16 und schließlich das Endekennzeichen mit der Priorität 0. Der Stapel ist anschließend leer, die Addition wird compiliert, und der Vorgang ist beendet. Der entstandene P-Code lautet »Zahl 8 Var A Sin Mult Zahl 16 Add«.

So einfach wie hier beschrieben, ist das Compilieren von Basic-Programmen jedoch nicht. Es sollte an dieser Stelle vielmehr gezeigt werden, wie der Turbo-Compiler prinzipiell funktioniert. Sollten alle Vorgänge beim Compilieren beschrieben werden, würde dies ein Buch füllen. Noch ein Tip: Das Standard-Atari-Basic ist wegen eines Betriebssystem-Fehlers nicht in der Lage, >A=NOT NOT A< zu berechnen. Dies hängt mit der Stapelverarbeitung zusammen. Probieren Sie diese Befehlsfolge doch einfach einmal auf Ihrem Atari 800XL aus. Aufgrund der Syntaxkontrolle ist der Befehl nicht zulässig. Die alten Atari-Computer reagierten auf »A=NOT NOT A« mit einem Systemabsturz. Geben Sie die Programmzeile doch einfach einmal unter Turbo-Basic-XL ein. Sie werden sehen, daß auch solche komplexen, logischen Verknüpfungen ordnungsgemäß funk-(Frank Ostrowski/wb)

## Bücher zum ATARI 65XE/800XL/130XE

I. M. Schreiber

#### Das Atan-Programmierhandbuch Márz 1985, 390 Seiten

BASIC lemen, seinen Atar: -BASIC lettient, seinen Ager – einschliebten Maschinensprache – bis ins Detail kennen-krieen und im Schluß arie Menge Programme haben, mit denen man vor seinen Freunden so richtig engeben kann (wessen Computer kann schon eine dreistimmige Fuge von J. S. Bach spisien?) Wenn Sie das wollen, dann ist dieses Buch für Sie gerade notwig, euch oder gerade wenn Sie Anlänger sind. Nach volliendeter Lekhore wird man das Buch aber nicht zur Seite legen; Es enthält eine solche Fulle an Informa-tionen, delt as such als Nechschlagewerk

bestens geeignet ist

BASIC und Attri-Einführung für den Anfanger, Nachschlagewerk für den Fortge-

Bent-Nr MT 753 ISBN 3-89090-062-3 DM 52,-4F4, 47,80/85 405,60





Poole/M McNiff/S. Cook Mein Atari-Comouter 1983, 500 Seiten

Ein Handbuch, das für jeden Atan-Bestzer wertvolle Informationen enthält. Ob Sie einen Asan 400, 800, 600 KL, 800 KL oder 1450 XLD bestzen, das Handbuch geh Ihnen Aus kunft zu allen Fragen der Bedienung und BASIC Programmierung, zum Umgang mit Kas-seiten, Drucker und Diskettenstation, zum Er-zeugen von Bild und Ton. Es ist reich bebildert und enthalt eine Vielzahl der für den ernsthaft Interessierten so wichtigen Tabellen Beson-ders im Anhang findet sich so mancher Lecker bissen für den ochten Freak, z.B. eine Erktärung der wichtigster Speicheradrassen für PEEKs und POKEs.

• En Handbuch zur Lösung aller Ateri-

Bost-Nr. PW 554 ISBN 3-921803-18-7 DM 59,-reFr. 54,30/53 460,20



#### H. L. Schneider/R. Bichler Das Atarl-Buch Band 1: Grundlegende Programmiermäglichkeiten Oktober 1984, 158 Seiten

Auch wer BASIC kann und sein Alari-Handbuch durch gearbeitet hat, wird ohne zu-sätzliche Anregungen seiner Atan-Comouter night völlig ausschöpfen können.

DM 32,-/sFr. 29,50/6S 249.60

#### Band 2: Spezielle Programmiermöglichkeiten und Maschinenprogramme Oktober 1984, 197 Seiten

Das ideale Buch für die Atan-Programmerung in Ma-schinensprache. Best-Nr. MT 704 DM 32,-/sFr. 29,50/6\$ 249,60



#### E H Carison Leme BASIC auf dem Atari November 1984, 325 Selten

Zu einem Computer für junge Leute gehört auch ein Lehr-und Lehrbuch für junge Leute. De tut es nicht irgendem BA-SIC-Buch, der Aufbau, die verwendeten Buspiele, das Lemtempo und nicht zuletzt der Ton müssen stimmen Nichts schreckt mehr ab als achauderhafter. Oberlehrer-Schulbuchstill Soldrehen sich auch die Beispiele vorzugs-weise um Spiele "Action-Brett- und Wortlästen), denn nuch an ihnen läßt sich Programmeren erlemen. Lektionen sind nicht Die Inktionen sind nicht zu lang, mit Grafiken aufgelockert und mit Üburiosbeisolelen verse hen (selbstverständlich inkur sive Lösungen)

Best-Nr MT 692 ISBN 3-89090-007-0 DM 38,-4:Fr 35,-65 296,40



#### H, Konl/T Kahrv L Lindsay/P Cleland Spiel und Spaß mit dem Atarl 1984, 338 Soites

Den Umgang mit dem Compu-tor lemt man am besten durch Ausprobieren, deshalb sofite ein Einführungsbuch mög-lichst viele Beispiele einfrah-ten. In diesem Buch finden Sie daher über 100 Pro-grannbausteine, die zusam-men mit 20 festi. granmbausteine, die zusam-men mit 20 fertigen und voll-standigen Programmen ein soldes Fundament für den Umgang mit BASIC und ihrem Computer legen, So können Sie besser die interessanten Ton- und Grafikmöglichkeiten des Alari nutzen.

• Ein Lehr und Arbeitsbuch

für Anlänger, die BASIG und ih ISBN 3-89090-002-X

DM 42,-/sFr. 38,60/6\$ 327,60



#### A. Z. Lamothe if Ausgesuchte Atan-Programme wit Listings Ottober 1984, 171 Seiten

Es gibt drei Möglichkeiten, mit einem Computer zu arbeiten man kann fertige Programme kaufen man iemt selber pro-grammieren oder tippt Pro-grammieren oder tippt Programme ab und macht sich so mit der Funktion seines Com-puters vertraut

Dieses Buch ist für diejenigen gedacht, die es mit der dritten Methode versuchen wollen Durch das Abschreiben der Durch das Abschreiben der Programme leinen Sie feich-ter den Umgang mit ihrem Computer und werden in die Grundbegriffe des Program-mierens herangeführt. Alle Programme läuden mit der Atlan-BASIC Version, die mit

allen Atan-Computern elefert wird Best-Nr MT 759

ISBN 3-89090-070-4



#### T Rowley

#### Sorühende Ideed mit Atan-Grafile Januar 1985, 224 Seiten

Ein eigenes Buch nur für Grafix? Bei der umwerfenden Grafik Mogiotikerten des Ata-nist es das schon wert Schließlich wollen Sie ja night nur in die Grundlagen der Bild-erzeugung eingeführt wer-den: Es soll schon auch bunt, bowegt und dreidmensional zugehen. Wie man Zeichen-sätze verändert erfahren Sie seibstverständlich auch. Ein billichen BASIC sollten Sie aber schon können, um mög-lichst viel von diesem Buch zu

Atan 400 oder 800. Best-Nr. PW 716 ISBN 3-921803-39-X DM 49,-IsFL 45,1065 382,20



#### H Gäcksman

#### Ber Atari als Musikhov November 1984, 184 Seiten

Die Soundmöglichkeiten des Atari sind schon beinaho legendlir, sie nutzen ihnen je-doch wenig, wenn Sie über die Prinzipien der Klanger zeugung und Musikprogram-mierung nicht Bescheid wis-sen. Deshalb bringt dieses Buch eine Fülle einfacher und lehrreicher Programmbei-spiele, die auch dem in BASIC nicht au Versierten das Ver ständnis der wichtigsten Techniken erlauben. Und so können Sie zuletzt nicht nur beliebige Range erzeugen oder den Computer in einen Synthesizer verwandeln, Sie besitzen auch eine eigene Diskothek« von bellebten und bekannten Melodien, die ihr Alan auf Knopfdruck vierstimmig spielen kann

Best.-Nr. MT 797 ISBN 3-89090-075-5 OM 29,80/sFr, 27,50/6\$ 232,40

#### Markt & Technik-Facilities erhalten Sie bei Ihrana Buchhündlar

Bestellkarten bitte an Ihren Buchhändier oder an einen unserer Depot-Händler. Adressenverzeichnis am Ende des Heftes.

Bestellungen im Ausland bitte an untenstehende Adressen. Schweiz. Merkt & Technik Vertriebs AG. Kollerstrasse 3, CH-6300 Zug, © 0.42/415656 Österreich: Rudolf Lechner & Sohn, Herzwerkstraße 10, A-1232 Wien, © 6222/877526



Kans-Pinsel-Straße 2, 8013 Haar bei München





## Der Unterschied liegt im Detail

Läuft Ihr Programm nicht unter Turbo-Basic XL? Einige Änderungen lassen auch Ihr normales Atari-Basic-Programm zum Renner werden.

urbo-Basic XL ist wesentlich schneller als das eingebaute Atari-Basic (siehe hierzu auch den Geschwindigkeitsvergleich zwischen den derzeit erhältlichen Basic-Versjonen für den Atari 800XL und 130XE in diesem Sonderheft). Die hohe Geschwindigkeit ist in den meisten Anwendungsfällen sicher erwünscht. Allerdings ergeben sich daraus auch manchmal Probleme. Oft sind nämlich in Atari-Basic geschriebene Spielprogramme unter Turbo-Basic XL kaum mehr spielbar, da die enorme Geschwindigkeit dem Spieler nicht mehr genügend Reaktionszeit erlaubt. Abhilfe: Verzögerungen in Form von Warteschleifen einbauen oder den speziellen PAUSE-Befehl verwenden. Oder bauen Sie doch einfach zusätzliche Geräusch-Effekte ein.

In Atari-Basic wird eine kurze Verzögerung auch mit »Q=1"1« realisiert. In Turbo-Basic XL wird diese Berechnung aber wesentlich schneller als in Atari-Basic ausgeführt. Potenzierung mit ganzzahligem Exponenten (<100) wird durch wiederholte Multiplikation berechnet, und nicht, wie in Atari-Basic, nach der Formel »a"b=exp(b\*log(a))«. Am einfachsten ist es, »Q=1"1« durch »PAUSE 9« zu ersetzen. Dies nimmt etwa die gleiche Zeit in Anspruch und spart sogar etwas Speicherplatz.

Verzögerungsschleifen mit FOR.. NEXT.. sind in Atari-Basic sehr langsam besonders wenn sie sich am Programmende befinden. Am besten ersetzen Sie solche Schleiten auch durch entsprechende PAUSE-Befehle.

#### Verbotene Befehle

Turbo-Basic XL beinhaltet wesentlich mehr Befehle als Atari-Basic. Deshalb können Probleme mit Variablennamen auftreten. Namen wie ERR, HEX\$ oder DEL(ta) sind nämlich in Atari-Basic erlaubt, in Turbo-Basic XL jedoch nicht (es handelt sich hierbei um reservierte Worte). Beim Laden und Starten eines Atari-Basic-Programms macht sich dieser Unterschied noch nicht bemerkbar. Erst wenn Sie versuchen, ein Programm mit diesen Variablennamen zu

editieren, ergeben sich Fehlermeldungen oder falsch übersetzte Programmzeilen. Bei »LET ERR=100« wird die Variable ERR benutzt, bei »A=ERR« jedoch die Turbo-Basic XL-Funktion »ERR«. Dann bleibt als einzige Alternative, die Variablennamen manuell zu ändern. Dabei hilft übrigens der DUMP-Befehl, mit dem man sich die im Programm verwendeten Variablen auflisten lassen kann.

Nach dem Laden des betreffenden Programms mit LOAD geben Sie einfach DUMP ein, und alle Variablen werden auf dem Bildschirm ausgegeben. Falls Sie einen Drucker besitzen, erfolgt die Ausgabe nach »DUMP "P: "« auf dem Drucker. Schreiben Sie sich dann alle Variablennamen auf, die geändert werden müssen. Dabei bleibt es Ihnen nicht erspart, jede einzelne Programmzeile zu kontrollieren und gegebenenfalls zu ändern. Speichern Sie das Programm anschließend mit »LIST "D:xxxxxxx" « auf Diskette, geben Sie NEW und dann »ENTER "D:xxxxxxx"« ein. Diese Befehlsfolge bewirkt, daß die Variablennamentabelle neu geschrieben wird. Nicht benutzte Variablen belegen dann auch keinen Speicherplatz mehr. Mit DUMP können Sie nochmals kontrollieren, ob Sie irgendwelche Variablen übersehen haben. Wenn Sie alle Korrekturen vorgenommen haben, sichern Sie das Programm mit SAVE auf Diskette und führen Sie einen Probelauf

Turbo-Basic XL liegt nicht im ROM vor, sondern wird von Diskette geladen. Da es noch wesentlich umfangreicher als Atari-Basic ist (etwa 18 KByte statt 8 KByte), wurde ein Großteil des Interpreters unter den OS-ROMs versteckt (12 KByte). Die restlichen 6 KByte befinden sich an der Speicheruntergrenze direkt über dem DOS.

Ursprünglich wurde Turbo-Basic XL auf einem Atari 400 mit 48 KByte RAM geschrieben. Bei dieser Version befand sich der gesamte Interpreter über dem DOS. Allerdings blieben so nur etwa 20 KByte für Basic-Programme übrig. Der 800XL besitzt dagegen insgesamt 64 KByte RAM, die Turbo-Basic XL auch voll nutzt. Von den zusätzlichen 14 KByte werden aber noch 2 KByte von Kopien der beiden ROM-Zeichengeneratoren belegt. Eine Maßnahme, die Bildstörungen bei Textdarstellung sobald das ROM abgeschaltet wird verhindert. Turbo-Basic XL ist nun aber viel zu umfangreich, um in die verbleibenden 12 KByte hineinzupassen. Deshalb ließ es sich nicht umgehen, auch einige Programmteile in Bereichen abzulegen, die normalerweise für Basic-Programme reserviert sind. Auf diese Art und Weise lassen sich Unterprogramme der ROMs benutzen. Weiterhin erlaubt dies noch die Durchführung der Interrupts.

Der im Basic-Bereich untergebrachte Teil von Turbo-Basic XL belegt die Adressen von \$2080 bis \$3629 (hexadezimal). Es handelt sich also um den Bereich direkt oberhalb von DOS 2.5 bei fünf installierten Diskettenlaufwerken (D1: bis D4: und der RAM-Disk D8: auf dem 130 XE) und 7 Filepuffern. Ein POKE in diesen Bereich dürfte meist zum Absturz des Computers führen, oder – was weitaus schlimmer wäre – zu einem Fehlverhalten des Interpreters. Denn daraus können sich sogar falsche Rechenergebnisse ableiten.

Während der Ausführung der Befehle POKE, DPOKE, MOVE und -MOVE wird jeweils der ROM-Bereich eingeschaltet. So können diese Befehle im »ROM«-Teil von Turbo-Basic XL keinen Schaden anrichten. Außerdem ist der ROM-Bereich noch bei den Befehlen PEEK, DPEEK, allen I/O-Befehlen wie PRINT, INPUT, BGET, BPUT etc. aktiviert, jedoch nicht bei PAINT, sowie der Ausführung einiger Arithmetik-Routinen (es werden einige Werte aus den Floating-Point-Routinen sowie die STR\$- und VAL-Routinen verwendet).

#### Vorsicht, Interrupt!

Tritt ein Interrupt auf, während die ROMs eingeschaltet sind, wird er wie gewohnt ausgeführt. Wenn die ROMs allerdings ausgeschaltet sind, verzweigt der Interrupt zuerst in den unteren RAM-Teil des Interpreters. Dort werden zuerst die ROMs eingeschaltet. Nachdem ein Interrupt für die ROM-Routinen simuliert wird, werden die ROMs wieder stillgelegt. Aus diesem Grund dauern die Interrupt-Routinen einige Mikrosekunden länger. Dies gilt allerdings nicht für den Display-List-Interrupt (DLI), da dieser parallel zum Bildaufbau ablaufen muß. Der Interrupt wird hier schon vom Interpreter erkannt, so daß er sofort ausgeführt werden kann. Er muß also auch bei abgeschalteten ROMs ablaufen.

Der Speicherplatz für Basic-Programme, Daten und den Stack reicht unter Atari-Basic bis \$A000 (40960). unter Turbo-Basic XL aber bis \$C000 (49152), Das Basic-ROM wird hierbei nämlich nicht benutzt und kann deshalb immer abgeschaltet bleiben. Reine Basic-Programme, also ohne USR-Aufrufe und POKEs, sind von dieser Verschiebung nicht betroffen. Kritisch sind POKEs und USR-Aufrute in Adressen im Bereich zwischen 30000 und 40960. Wenn Sie also ein Programm verwenden, das einen Befehl wie beispielsweise »POKE 39994,4« enthält, dürfte das Programm unter Turbo-Basic XL nur mit einigen Anpassungen laufen. Dieser POKE-Befehl verändert übrigens die Display-List. Hierbei handelt es sich um die Maschinensprache des Videoprozessors, der Antic genannt

Da es auch noch den 600 XL mit 16 KByte und den Atari 400 und 800 mit 16, 32 und 48 KByte RAM gibt, existleren auch Programme, die sich an verschiedene Speichermengen anpassen. Dazu benutzt das OS (Operating System) des Atari verschiedene Pointer (Zeiger), auf die auch Basic zurückgreift. Der wichtigste davon ist RAM-TOP, also die Speicherzelle 106 (\$6A).

Sie enthält das MSB (Most Significant Byte) der ersten nicht mehr benutzbaren Speicherzellen für Programme. Das heißt, nach »PEEK(106)\*256« erhält man einen Wert, der um eins höher ist als die hochste RAM-Adresse. In Atari-Basic ergibt »PEEK(106)« also 160 (\$A0), in Turbo-Basic XL entsprechend 192 (\$C0). Durch »POKE 106,PEEK (106)-16« lassen sich so 4096 Bytes (16\*256) vor dem Zugriff durch das OS und Basic schützen.

Anschließend ist noch ein GRA-PHICS-Befehl nötig, um die anderen Pointer anzupassen und um den Bildschirmspeicher zu verschieben. Durch »POKE 106,160,GRAPHICS 0« am Anfang eines Programms wird also die Atari-Basic-Speicheranordnung simuliert. Dann bleiben allerdings wieder 8 KByte weniger Speicherplatz für das Basic-Programm übrig. Dafür beginnt aber der Bildschirmspeicher schon ab Adresse 40000 statt 48192. Für kurze Programme reicht diese Minimalanpassung vollkommen aus. Sollte ein Programm aber länger sein, wie zum Beispiel »Magic-Painter« (Listing des Monats, Happy-Computer, Ausgabe 3/85), gibt es zwei Möglichkeiten:

Adresse Turbo-Basic XL Adresse Atari-Basic **FFFF** FFFF OS-ROM OS-ROM Interpreter D800 D800 I O-Chrps I/O-Chips D000 D000 OS-ROM OS-ROM Interpreter C000 C000 Grafik-RAM Basic-ROM A000 Grafik-RAM Basic-Stack Arrays Basic-Stack Programm Arrays 3629 Programm Interpreter 2080 2080 DOS 2.5 DOS 2 5 0700 0700 OS-Variablen OS-Variablen 0000 0000

Die Speicheraufteilung von Turbo-Basic XL im Vergleich zum normalen Atari-Basic.

1) Alle POKE- und USR-Adressen um 8192 erhöhen. Dies klappt aber nur bei relokatiblen, das heißt, bei verschiebbaren Daten und Maschinenprogrammen. Leider sind diese recht selten; und wenn sie autreten, dann hat der Programmierer das Maschinenprogramm meist in einem String untergebracht (das Kennzeichen dafür ist stets: »USR(ADR(X\$),...))«. Dieser Aufruf bereitet aber keine Schwierigkerten. Sie können natürlich die Maschinenprogramme relozieren, also an die andere AdreBlage anpassen. Dies erfordert aber Assemblerkenntnisse und viel Zeit- und Arbeitsaufwand. Einfacher ist es, wenn der Quelltext des Programms zur Verfügung steht.

#### Verbotene Adressen

Auf Probleme stößt man auch bei den Daten für Player/Missile-Grafiken oder bei veränderten Zeichensätzen. Hier sind die POKE-Befehle, die deren Adressen dem Operating System oder der Hardware mitteilen, anzupassen. Dabei handelt es sich meist um POKEs auf die Adressen 756, 54279, 561 und 89 (und 513 etc.). Der gePOKEte Wert ist dann jeweils um 32 zu erhöhen. Manchmal werden diese Adressen auch innerhalb von Maschinenprogrammen angesprochen.

Sie sehen also, daß diese Anpassung ohne genaue Kenntnisse des Programms und der Hardware fast unmöglich ist

Das Programm um 8 KByte kürzen. Dies ist bei großzügig (gut) kommentierten Programmen manchmal allein durch Streichen der REM-Zeilen zu erreichen. Oft hilft auch das Auslagern von DATAs in ein Vorprogramm, das in einer READ-Schleife Daten einPOKEt, und dann das gekürzte Hauptprogramm mit »Run "D.xxx.xxxx"« startet. Schneller ist aber ein entsprechender BLOADoder BGET-Befehl, wenn die Daten im entsprechenden Format auf Diskette vorliegen. Dies spart übrigens auch wertvollen Platz auf der Diskette. Weiterhin läßt sich auch durch geschickten Einsatz der erweiterten Turbo-Basic XL-Befehle viel Speicherplatz sparen.

Durch einen kombinierten Einsatz dieser Mittel wurde auch das Programm »Magic-Painter« angepaßt. Besonders hilfreich waren dabei die Befehle BLOAD, CIRCLE und TEXT. Noch ein Tip: Wäre »Magic-Painter« in Turbo-Basic XL geschrieben worden, könnte es ohne weiteres in der höchstauflösenden Farbgrafikstufe 15+16 arbeiten, statt in Grafikstufe 7+16. Schließlich stehen unter Turbo-Basic XL 8 KByte RAM mehr zur Verfügung.

(Frank Ostrowski/wb)



## Auf die Taste, fertig, los!

m Aussagen über die Geschwindigkeit eines Computers oder einer Programmiersprache machen zu können, bedient man sich üblicherweise sogenannter Benchmark-Programme. Hierbei handelt es sich um kurze Algorithmen, die möglichst ohne Änderung auf allen zu testenden Versionen laufen sollten. In diesem Fall wurden alle Benchmarks in Standard-Atari-Basic geschrieben und, soweit dies durchführbar war, direkt in die anderen Sprachen übernommen.

Neun unterschiedliche Benchmarks

Wie schnell Turbo-Basic-XL und der Turbo-Basic-Compiler wirklich sind, soll ein Vergleich mit anderen Programmiersprachen und Compilern zeigen.

werden erweisen, wie schnell Turbo-Basic XL und der Turbo-Compiler im Vergleich mit anderen Programmiersprachen für den Atari sind. Zur Zeitmessung wurden die Speicherstellen 19 und 20 herangezogen. Die Speicherstelle 20 wird durch einen Interruptimpuls 50mal pro Sekunde inkrementiert.

Ist der Wert 256 erreicht, erhöht sich der Wert in Speicherstelle 19 um 1 und Adresse 20 wird auf 0 zurückgesetzt. Liest man diesen Zwei-Byte-Wert und teilt ihn durch den Faktor 50, so erhält man eine Zeitangabe in Sekunden. Um die Unterschiede der Rechenzeiten deutlicher zu machen, wird jeder Benchmark 1000mal durchlaufen.

Als erste Vergleichssprache diente natürlich das normale Atari-Basic. Wie man der Tabelle entnehmen kann, ist

```
10 DIM M(5) -A$ (1900)
                                            <CC>
                                                   618 K=0
                                                                                                 (FT)
20 ? "ATARI_BENCHMARKS"
                                             <CH>
                                                    620 K=K+1
                                                                                                 <RH>
                                                   638 A=K/2#3+4-5
99 REM *** Bench 1 ***
                                             (PA)
                                                                                                 <ZK>
100 POKE 20,0:POKE 19,0
                                             < XQ>
                                                    648 60SUB 2000
                                                                                                 < RG >
110 FOR I=1 TO 1000
                                            CGVS
                                                    650 FOR L=1 TO 5
                                                                                                 CPD /
120 NEXT I
                                            <EV>
                                                                                                 < HM >
130 T1=PEEK(20)+PEEK(19)*256
                                             (CE>
                                                   670 IF K<1000 THEN 620
                                                                                                 STU
190 7 "T1_=_"; T1/50
                                            <67>
                                                    680 T6=PEEK (20) +PEEK (19) +256
                                                                                                 CDXS
                                                   690 ? "T6_=_":T6/50
699 REM *** Bench 7 ***
199 REM *** Bench 2 ***
                                            <0Z>
                                                                                                 LZZ
200 POKE 20.0: POKE 19.0
                                             <XR>
                                                                                                 SV.
                                             <FP>
210 K=0
                                                   700 POKE 20,0:POKE 19,0
                                                                                                 VM.
220 K=K+1
                                             (RD>
                                                   710 K=0
                                                                                                 KEU
230 IF K<1000 THEN 220
                                             < DG >
                                                   720 K=K+1
                                                                                                 (RI)
240 T2=PEEK(20)+PEEK(19) *256
                                             <EN>
                                                   730 A=K/2#3+4-5
                                                                                                 · ZL >
290 ? "T2.=.";T2/50
                                            <HZ>
                                                   740 BOSUB 2000
                                                                                                 RH.
299 REM *** Bench 3 ***
                                             <PT>
                                                                                                 KPE /
                                                   750 FOR L=1 TO 5
300 POKE 20,0: POKE 19,0
                                             <XS>
                                                   755 M(L)=A
                                                                                                 < BM .
310 K=0
                                             (FQ)
                                                                                                 < HN
                                                   760 NEXT L
                                             <RE>
328 K=K+1
                                                   778 IF K<1000 THEN 720
                                                                                                 UF /
330 A=K/K*K+K-K
                                             <0H>>
                                                   780 17=PEEK (20) +PEEK (19) *256
                                                                                                 SEE /
340 IF K<1000 THEN 320
                                            <RD>
                                                   790 ? "T7_=_"; 77/50
                                                                                                mZ /
350 T3=PEEK (20)+PEEK (19) *256
                                            <CW>
                                                   799 REM *** Bench 8 ***
                                                                                                 TF >
390 7 "T3_=_":T3/50
                                             <IZ>
                                                   800 POKE 20,0: POKE 19,0
                                                                                                - X X
399 REM *** Bench 4 ***
                                             < DN>
                                                   810 K=0
                                                                                                 FV
400 POKE 20,0:POKE 19,0
                                             <XT>
                                                   820 K-K+1
                                                                                                 (RJ>
                                             <FR>
410 K=0
                                                   830 A=K^2
                                                                                                 <FO>
420 K=K+1
                                             (RE)
                                                   840 B-LCG(K)
                                                                                                 < WU>
438 A=K/2#3+4-5
                                             <ZI>
                                                   850 C=SIN(K)
                                                                                                 (ZN)
440 IF K<1000 THEN 420
                                                   860 IF K<1000 THEN B20
                                             <RY>
                                                                                                 (VI)
450 T4=PEEK(20)+PEEK(19) *256
                                             <000>
                                                   870 T8=PEEK(20)+PEEK(19) *256
                                                                                                 <EJ>
490 ? "T4_=_"; T4/50
                                             <3Z>
                                                   890 ? "TB_=_"; T8/50
                                                                                                 (NZ)
499 REM *** Bench 5 ***
                                             <RH>
                                                   899 REM ***BENCH 9 ***
                                                                                                 <KG>
500 POKE 20,0: POKE 19,0
                                             <XU>
                                                   700 POKE 20,0: POKE 19,0
                                                                                                < XY>
510 K=0
                                             (ES)
                                                   910 FOR 1=1 TO 1000
                                                                                                 < HD>
520 K=K+1
                                             <RG>
                                                   720 A$(I,I)="A"
                                                                                                < WZ >
530 A=K/2*3+4-5
                                             <ZJ>
                                                                                                SEX
                                                   938 NEXT I
540 GOSUB 2000
                                            (RE)
                                                   940 T9-PEEK (20) +PEEK (19) +256
                                                                                                 <EN
                                                    950 ? "T9_=_";T9/50
550 IF K<1000 THEN 520
                                             (SV)
                                                                                                 COR >
                                                    1000 TS-T1+T2+T3+T4+T5+T6+T7+T8+T9
560 T5=PEEK(20)+PEEK(19) *256
                                             < BMD
                                                                                                 MR L>
590 ? "T5_=_";T5/50
                                             <KZ>
                                                    1010 ? :? "SUM-__";TS/50
                                                                                                 AHA .
599 REM *** Bench 6 ***
                                             (SB)
                                                    1020 END
                                                                                                FV
600 POKE 20,0: POKE 19,0
                                                   2000 RETURN
                                                                                                 - 00
```

Benchmark: Nummer	Atari- Basic	Turbo- Basic	Basic XE normal	Basic XE fast	Turbo- Compiler	Action	MMG- Compiler	ABC- Compile
T1	2.22	0.78	1.50	1.42	0.50	0.02	0.92	0
T2	7.64	3.06	4.56	3.04	0.54	0.02	1.14	1
T3	20 24	7.58	16 08	8.82	2 82	0.82	8 38	5
T4	24.12	8.62	16.66	9.72	3.78	16.14	11.86	5
T5	33.74	9.08	25.92	17.62	3.79	16.14	11.86	8
TØ	58.30	14.54	45.40	37.12	7 00	16.26	16.44	10
T7	83 58	24 86	62 28	49.98	11.78	17 86	22.73	13
T8	425.34	56.02	419 80	59 28	50 30	467 44	417 22	_
T9	8 14	3 24	6 88	4 20	1 24	0 02	2 68	1
T1-T7	229 84	88.50	172 40	127.72	30 38	67 28	73.32	40
Summe	663 32	127.76	559 08	191.20	81 92	534.76	493 22	_

Tabelle der erzielten Rechenzeiten

Turbo-Basic XL bei den einzelnen Tests um den Faktor 2 bis 7 schneiler als Atari-Basic. Der Turbo-Compiler erweist sich sogar als rund achtmal so schnell wie normales Basic.

Basic XE schneidet in diesem Vergleich schon wesentlich besser ab. Im normalen Modus ist es zwar kaum schneller als Atari-Basic, im sogenannten »FAST«-Modus jedoch kommt es fast an die Zeiten von Turbo-Basic XL

heran. Hierbei muß allerdings erwähnt werden, daß sowohl in Basic XE wie auch in Turbo-Basic XL durch die Verwendung ihrer spezifischen Befehle eine Geschwindigkeitserhöhung erzielbar ist.

In »Action«, einer Compilersprache, konnte das Basic-Listing natürlich nicht mehr übernommen werden. Jedoch wurden die sinngemäß gleichen Algorithmen verwendet. Lediglich bei Benchmark 8 wurde die Sinus-Funktion durch den natürlichen Logarithmus ersetzt, da in »Action« die Sinusfunktion nicht implementiert ist und daher nur durch Reihenentwicklung ausführbar wäre. Wie die Tabelle zeigt, ist »Action« bei der Festkommarechnung kaum zu schlagen. In der Gleitkommarechnung jedoch (ab Benchmark 4) läßt es aber deutlich nach

Um einen direkten Vergleich zwischen Basic-Compilern zu erhalten. wurden auch der MMG-Compiler und der ABC-Compiler in die Tabelle aufgenommen. Wie man sieht, ist der Turbo-Basic-Compiler in jedem Punkt schneller als der MMG-Compiler. Der Vergleich mit dem ABC-Compiler hinkt jedoch ein wenig. Dieser Compiler kann nämlich keine Gleitkommazahlen verarbeiten. Daher sind alle Ergebnisse mit Festkommazahlen berechnet. naturgemäß sind die Zeiten entsprechend besser So erklären sich auch die gerundeten Zeitangaben in der Tabelle. Benchmark Nummer acht konnte mit dem ABC-Compiler nicht bearbeitet werden, da weder die Sinus- noch die Logarithmus-Funktion compilert wer-

(Wolfgang Czerny/wb)





# Basic, schnell wie der Wind – mit dem Turbo-Basic XL-Interpreter

Ist Ihnen Atari-Basic zu langsam? Oder wünschen Sie sich zusätzliche Befehle? Dann ist unser Turbo-Basic XL interpreter genau das richtige.

ugegeben, unser Interpreter ist ein wenig umfangreich. Immerhin müssen insgesamt 18108 Byte erst einmal eingetippt sein. Eine Aufgabe, die sicher einige Stunden in Anspruch nimmt. Haben Sie aber dann den Interpreter auf Diskette vorliegen, sparen Sie sich zukünftig viel Zeit für die Entwicklung neuer Programme. Erstens stehen Ihnen sämtliche Befehle des normalen Atari-Basic zur Verfügung, und zweitens sparen die neuen Befehle viel Zeit und vor allem Speicherplatz.

Aber Turbo-Basic XL spart nicht nur Zeit bei der Programmierung, sondern auch bei der Ausführung von Programmen. Es macht seinem Namen als schnellstes derzeit erhältliches Basic für Atari-Computer nämlich alle Ehre. Wir haben unseren Interpreter dem normalen Basic und verschiedenen Basic-Compilern gegenübergestellt. Turbo-Basic XL siegte

In jeder Hinsicht.

Damit Sie mit Turbo-Basic XL und dem dazugehörigen Compiler arbeiten können, benötigen Sie unbedingt einen Atari 800XL oder 130XE. Besitzer der älteren Computer, also Atari 400 und 800, können den Interpreter leider nicht benutzen. In der abgedruckten Version setzt Turbo-Basic XL mindestens 64 KByte-RAM voraus. Damit aber alle die Vorzüge von Turbo-Basic nutzen können, befindet sich auf der Leserservice-Diskette eine spezielle Version für die älteren Atari-Modelle.

Beachten Sie bitte: Turbo-Basic XL muß unbedingt mit AMPEL (Atari-Maschinen-Programm-Eingabe-Hilfe) eingegeben werden. Sie finden es auf Seite 87. Von Basic aus läßt sich Turbo-Basic XL nicht eingeben. Wenn Sie alle dazugehörigen Erklärungen befolgen, kann nichts schiefgehen.

#### Besondere Hinweise zu Turbo-Basic XL

Bevor Sie Turbo-Basic XL eintippen, müssen Sie AMPEL (Atari-Maschinen-Programm-Eingabe-Hilfe) eingeben.

Turbo-Basic XL ist als »AUTORUN.SYS«-File ausgelegt. Das heißt, wenn Sie den Interpreter komplett eingegeben haben, können Sie Ihrem File einfach den Namen »AUTORUN.SYS« geben. Nach dem Einschalten des Computers wird dann automatisch Turbo-Basic XL geladen. Anschließend können Sie auch schon mit dem Interpreter arbeiten. Einzige Voraussetzung. Die Files »DOS.SYS« und »DUP.SYS« (entweder DOS 2.0 oder 2.5) müssen sich auf der gleichen Diskette befinden. Mit DOS 3.0 arbeitet Turbo-Basic XL übrigens nicht!

Hier noch einige Hinweise, wie Sie einen Datemamen auf Diskette ändern können

Nehmen wir an. Sie haben Ihrem Interpreter bei der Eingabe mit AMPEL den Namen »TURBO« gegeben. Dieses File soil nun den Namen »AUTORUN.SYS« erhalten. Entfernen Sie also zunächst Ihre Turbo-Basic XL-Diskette, und schalten Sie bitte den Computer aus. Nachdem Sie eine formatierte Diskette, auf der sich die Files »DOS.SYS« und »DUP.SYS« befinden, ins Laufwerk gelegt haben, schalten Sie den Computer wieder ein. Anschließend geben Sie »DOS« ein und gelangen über die RETURN-Taste ins DOS-Menü. Nun rufen Sie auf der Turbo-Basic XL-Diskette mit »E« die Funktion »Rename File« auf und drücken RETURN. Die Meldung »Give - Old Name, New Name« erscheint. Geben Sie beispielsweise »TURBO, AUTORUN. SYS« ein. Sollte Ihre Datei einen anderen Namen haben, müssen Sie diesen anstelle von »TURBO« einsetzen. Nach Betätigung der RETURN-Taste wird Ihr File dann umbenannt und hat dann den Namen »AUTORUN.SYS«. Überzeugen Sie sich dann, daß der soeben beschriebene Vorgang auch wirklich funktioniert hat. Mit »A« für Directory und RETURN wird das neue Directory auf dem Bildschirm angezeigt. Folgende Programme sollten sich letzt auf Ihrer Diskette befinden: DOS.SYS, DUP.SYS und AUTORUN SYS.

Noch eins, bevor Sie mit Turbo-Basic XL arbeiten, sollten Sie sich mindestens eine Sicherheitskopie Ihrer Diskette anlegen. Nehmen Sie sich also eine neue oder eine Diskette, die Sie löschen können, und formatieren Sie diese. Falls Sie über DOS 2.5 verfügen, können Sie sich das Formatieren sparen. Mit der Funktion »J« im DOS-Menü wird nun die Diskette kopiert. Wenn Sie über ein Diskettenlaufwerk verfügen, geben Sie jetzt »D1,D1« ein und betätigen daraufhin die RETURN-Taste. Der Kopiervorgang beginnt. Dabel werden Sie mehrmals aufgefordert, die Original-Diskette mit der Duplikat-Diskette auszutauschen.

Sie haben jetzt also zwei Disketten vorliegen, die exakt die gleichen Programme enthalten. Arbeiten Sie aber bitte nur mit einer der Disketten, und legen Sie die andere an eine gleichere Stelle

sichere Stelle.

Nach dieser Sicherheitsmaßnahme können Sie endlich Turbo-Basic XL ausprobieren. Dazu muß sich nur ihre Interpreter-Diskette im Laufwerk befinden. Schalten Sie jetzt die Stromversorgung Ihres Computers aus und wieder ein. Wenn

Sie dabei keine Taste, auch nicht die OPTION-Taste, mit der normalerweise das eingebaute Basic abgestellt wird, betätigt haben, erscheint das Titelbild und kurz danach die READY-Meldung. Sie befinden sich jetzt in Turbo-Basic XL. Viel Spaß

beim Programmieren.

```
250 END
293
1000 PROC RECHNEM
1010 GRAPHICS 15+16:POKE 764,255
1030 DX-(XMAX-MHIM)/159
1040 DY-(YMAX-MHIM)/159
1040 DY-(YMAX-MHIM)/191
1050 CX-MHIM:CY-YMAX
1050 FOR ZEILE-0 TO 191
1090 FOR SPALTE-0 TO 159
1110 HILE TI(TMAX AND (X2+Y2)(8
1130 YMC2*XMMYW-CY-XMCX2-Y2-CX
1130 YMC2*XMMYW-CY-XMCX2-Y2-CX
1130 MEND
1190 IF TI=TMAX
1200 CLOR X0
1220 ELSE
1230 COLOR X1 MOD 33+X1
1240 ENDIF
1250 CX-CX+DX:PLOT SPALTE,ZETLE
1260 IP PEEK(764)=20 THEN POP:GOT
0 1295
1276 MEXT SPALTE
```

Ein typischer Turbo-Basic XL-Bildschirm



#### Strukturierte Programmierung

In der folgenden Beschreibung bedeuten: aexp=arithmetic expression (arithmetischer Ausdruck) sexp=string expression (Text Ausdruck (A\$, "TEXT", CHR\$(), STR\$(), HEX\$())

lineno=Zeilennummer

»...«= Ein oder mehrere Befehle. In einer IF-THEN-Abfrage handelt es sich hierbel um den nach THEN folgenden Text. Sonst gibt es keine Beschränkung, »...« kann also auch mehrere Programmzeilen und/oder den Teil einer Programmzeile umfassen.

Es gibt im Standard-Atari-Basic nur zeilennummerorientierte Sprungbefehle und die »FOR...NEXT«-Schleife zur Konstruktion von Schleifen. In Turbo-Basic XL gibt es zusätzliche. an Pascal angelehnte, Strukturelemente.

IF aexp THEN lineno

IF aexp THEN ...

Das normale IF...THEN Statement.

IF aexp ... ENDIF

IF aexp ... ELSE ... ENDIF

Wenn die Bedingung aexp erfüllt (< > 0) ist, wird der Programmteil zwischen IF und ELSE, sonst der zwischen ELSE und ENDIF, ausgeführt. ELSE kann auch entfallen. Als Trennzeichen nach aexp dient nicht THEN, sondern ein Doppelpunkt (:) oder Zeilenende (RETURN). Auch vor und nach ELSE und ENDIF muß ein solches Trennzeichen stehen. Nach ELSE darf keine Zeilennummer folgen. In Atari-Basic benötigt man dafür oft mehrere GOTO-Befehle.

REPEAT ... UNTIL aexp

Wiederholt die Anweisungen »...«, bis die Bedingung erfüllt ist. Hier wird die Bedingung erst am Ende der Schleife geprüft. Der Programmteil zwischen REPEAT und UNTIL wird also mindestens einmal ausgeführt

WHILE aexp ... WEND

Wiederholt »...«, solange die Bedingung erfüllt ist, das heißt, wenn die Bedingung das erste Mal nicht erfüllt ist, wird die Schleife kein einziges Mal ausgeführt.

DO ... LOOP

Endlosschleife. Wiederholt die Anweisungen »...« immer wieder von neuem.

#### EXIT

Verläßt eine Schleife, Sprung ans Schleifenende. Dieser Befehl ist verwendbar bei »DO ... LOOP« und »REPEAT ... UNTIL«, »WHILE ... WEND« und auch bei »FOR ... NEXT«-Schleifen. Dieser Befehl stellt eine Art Notausgang aus Schleifen dar, bei »DO ... LOOP« sogar den einzigen, der in strukturierten Programmen erlaubt ist. Es wird stets an das Ende der Schleife gesprungen. Eine Schleife läßt sich auch mit »POP:GOTO lineno« abbrechen. Dies sollte jedoch nur im Notfall geschehen, da dieser Befehl nicht zur übersichtlichen Programmierung beiträgt.

★F

\*F +

Nach diesem Befehl sind »FOR ... NEXT«-Schleifen abweisend. Es wird also vor dem ersten Durchlauf der Schleife geprüft, ob der Zähler schon den Endwert erreicht hat. Dazu ein Beispiel: »FOR I=2 TO 1:? I:NEXT I«.

- Der Anfangswert der Schleife ist 2
- Schleifenzähler um 1 erhöhen
- Vergleich mit dem Endwert (1)
- Verlassen der Schleife, da die Bedingung erfüllt ist.

Nach ★F + wird zuerst die Variable I mit dem Anfangswert 2 geladen und dann mit der Variablen I (Endwert=1) verglichen. Anschließend wird die Schleife bis zum »NEXT k-Befehl übersprungen. Es erfolgt keine Ausgabe auf dem Bildschirm. Der »F+«-Befehl erspart oft eine spezielle Abfrageroutine.

★F-

Stellt den Normalzustand wieder her. »FOR ... NEXT«-Schleifen werden mindestens einmal durchlaufen. Auch bei RUN wird automatisch ein » ★ F -« ausgeführt (entspricht dem Atari-Basic).

PROC name

Beginn eines Unterprogramms (Prozedur) mit dem Namen name.

#### **ENDPROC**

Prozedurende, Entspricht dem RETURN nach einem »GOSUB lineno«,

**EXEC** name

Ruft die Prozedur name auf, entspricht »GOSUB lineno«. Die normalen Befehle GOSUB und RETURN erlauben, nur Unterprogramme mit einer Zeilennummer aufzurufen. Hier ist der Aufruf mit einem leichter zu behaltenden Namen möglich. Außerdem belegt EXEC-PROC-ENDPROC meist sogar weniger Speicherplatz als GOSUB-RETURN und lat etwas schneller. Die Prozedurnamen werden genauso gespeichert wie die Variabiennamen. Jeder Name belegt 8 Byte plus ein Byte für jedes Zeichen. Jede weitere Verwendung benötigt dann nur ein (zwei) Byte. Die Zeilennummer einer GOSUB-Anweisung belegt dagegen 7 Byte. Obwohl Turbo-Basic XL kompatibel zu Atari-Basic ist, können statt 128 jetzt 256 verschiedene Variablen- oder Prozedurnamen verwendet werden. Ab der 129sten Variablen kostet jede Verwendung zwei, statt bisher nur 1 Byte Speicherplatz.

ON aexp EXEC pname,pnam...

Entspricht der Atari-Basic-Anweisung »ON GOSUB«. pnam = PROCedurname.

# name

GO# name

ON aexp GO # nam.nam.nam

TRAP #name

#### RESTORE # name

Labeldefinition, das # entspricht dem PROC. In Turbo-Basic XL läßt sich der unstrukturierte Sprungbefehl GOTO wenigstens dadurch lesbarer machen, daß »GOTO zeilennummer« durch »GO # name« ersetzt wird. Auch bei TRAP und RESTORE können Marken verwendet werden. Dabei wird die Kennung » # « verwendet, um dem Interpreter mitzuteilen, daß keine Zeilennummer, sondern ein Name folgt. Ubrigens ist »GO# name« schneller als »GOTO zeile«. Die Marken für GO#, TRAP# und RESTORE# werden mit # gekennzeichnet, die für EXEC mit PROC. Diese strikte Trennung dient zur Erhöhung der Gliederung und Transparenz von Programmen.

POP

POP gilt für Unterprogramme mit GOSUB und EXEC ebenso wie für die Schleifen »FOR ... NEXT«, »REPEAT ... UNTIL«, »WHILE ... ENDWHILE« und »LOOP ... ENDLOOP«. Auf dem Runtimestack belegen GOSUB, EXEC, REPEAT, WHILE und LOOP jeweils 4 Byte (wie GOSUB in Atari-Basic). »FOR ... NEXT« belegt jedoch 13 statt 12 Byte, da jetzt 256 verschiedene Variablen zur Verfügung stehen. Auf dem Stack werden jetzt aber nicht mehr die Zeilennummern gespeichert, ein Grund für das langsame Atari-Basic, sondern die Speicheradressen dieser Zeilen. Dadurch werden Schleifen schneller. Außerdem sinkt die Geschwindigkeit der Schleifenwiederholung nicht mit steigender Entfernung vom Programmanfangl Trotzdem konnte einer der Vorzüge von Atari-Basic beibehalten werden: Ein Programm, das beispielsweise durch einen Programmfehler angehalten wurde, kann man editieren und dann mit CONT oder GOTO fortsetzen. Dabei werden weder die Variablen noch der Stack gelöscht. So lassen sich auch Programme schreiben, die ihre DATA-Zeilen selbst generieren oder nicht mehr benötigte Programmteile löschen (mit POKE 842, 13, danach verhält sich der Computer so, als würde ständig RETURN

gedrückt, bis POKE 842,12 den Normalzustand wieder herstellt). Zum Löschen von Programmteilen gibt es in Turbo-Basic XL den DEL-Befehl.

Dies ist eine spezielle REM-Anweisung. Bei der Eingabe einer Programmzeile wird alles nach »- « nicht beachtet. Bei LIST werden aber nicht zwei, sondern 30 Minuszeichen gedruckt. Diese Anweisung belegt sogar ein Byte weniger Speicherplatz als ein REM ohne Text. Eine entsprechende REM-Zeile im normalen Basic benötigt wesentlich mehr Speicherplatz.

LIST

Beim LISTen von Programmen werden Schleifen optisch durch Einrücken um jeweils zwei Leerzeichen hervorgehoben. Programme sind so übersichtlicher und strukturierter. Außerdem lassen sich auf diese Art und Weise manche Fehler vermeiden. Bei unsauberer Programmierung (mehrere NEXT zu einem FOR, mehrere ENDPROCs zu einem PROC...) gerät das Listing, ebenso wie der Interpreter, allerdings in Unordnung. Solche Konstruktionen, die in Atari-Basic und vielen anderen Basic-Versionen nur schwer zu umschreiben sind, können in Turbo-Basic XL leicht durch IF-ELSE-ENDIF oder EXIT ersetzt werden. Daraus resultieren wieder leichter lesbare Programme.

+ L -

schaltet die Tabulierung ab. Dies kann notwendig sein, um lange Programmzeilen zu editieren oder um Platz beim Speichern auf Diskette (LIST"D:X") zu sparen. Die Trennzeile » - « wird nach » \* L - « nur als doppeltes, statt dreißigfaches Minus-Zeichen gelistet.

**★**L +

schaltet die Tabulierung wieder ein (Normalzustand nach dem Laden des Interpreters). Übrigens gibt es jetzt auch ein LIST von einer Programmzeile bis zum Programmende. »LIST 3000, «Ilstet ab Zeile 3000 bis zum Programmende, »LIST "P:",3000, «entsprechend auf den Drucker (das einzelne Komma», «nach der Zeilennummer veranlaßt den Computer, automatisch 32767 für die höchstmögliche Zeilennummer zu ergänzen).

#### Neue Fehlermeldungen

Es gibt folgende neue Fehlermeldungen:

**ERROR - 22 ?NEST** 

Schachtelungsfehler, tritt auf, wenn das zu einem WHILE gehörende ENDWHILE nicht gefunden wird, oder das ENDIF zu einem IF, oder auch, nach »★F +<, das NEXT zu einem FOR. Beim Verlassen von Unterprogrammen (durch RETURN oder ENDPROC) werden Schleifen abgebrochen. Dies gilt sowohl – wie gewohnt – für die »FOR ... NEXT<-Schleife, wie auch für die anderen, unter Turbo-Basic XL zur Verfügung stehenden Schleifen.

**ERROR - 16 ?GOSUB** 

Zu einem GOSUB fehlt RETURN.

ERROR - 13 ?FOR

Zu einem NEXT fehlt FOR.

**ERROR - 23 ?WHILE** 

Zu einem WEND fehlt WHILE

**ERROR - 24 ?REPEAT** 

Zu einem UNTIL fehlt REPEAT.

**ERROR - 25 ?DO** 

Zu einem LOOP fehlt DO.

**ERROR - 28 ?EXEC** 

Zu einem ENDPROC fehlt EXEC.

ERROR - 29 ?PROC

Eine unbekannte Prozedur wurde aufgerufen.

ERROR - 30 ?#

Eine unbekannte Marke wurde verwendet.

#### **ERROR - 27 XPROC**

(Executing PROC). Diese Fehlermeldung tritt auf, wenn eine PROC-Anweisung ausgeführt wird. Prozeduren dürfen nur von EXEC aufgerufen werden.

**ERROR - 26 ?EXIT** 

EXIT ohne Schleife.

#### **ERROR - 15 ?DEL**

Das GOSUB zu einem RETURN, NEXT zu einem FOR. REPEAT zu einem UNTiL ... wurde gelöscht. In Atari- und in Turbo-Basic XL lassen sich Programme editieren, ohne Variablenwerte oder den Stapel zu zerstören. Wenn dann bei Rückkehr aus einem Unterprogramm (Schleife) die entsprechende Zeile gelöscht oder verändert wurde, kann dieser Fehler passieren. Dies tritt auch auf, wenn ein in ein Programm eingebautes DEL sich selbst löscht. Alle Fehlernummern, die Atari-Basic ohne irgendwelche Texte ausgibt, werden in Turbo-Basic XL grundsätzlich mit einem kurzen Text ergänzt (beispielsweise »138 TIMEOUT«, »29 ?PROC« etc.). Ausführlichere Erläuterungen können dem Atari-Basic Referenz Manual oder der DOS-Anleitung entnommen werden. Längere Texte würden, bei den insgesamt 60 zur Verfügung stehenden Fehlern, den Platzbedarf des Interpreters noch wesentlich erhöhen.

**DEL** von.bis

Löscht die Programmzeilen von-bis (jeweils einschließlich).

**RENUM** alt.neu.incr

Numeriert alle Programmzeilen ab Zeile alt um. Die neuen Zeilennummern beginnen bel neu und werden jeweils um increrhöht. Alle Zeilennummern vor alt bleiben unverändert. Dieser Befehl ändert auch die Zeilennummern nach GOTO, GOSUB, TRAP, RESTORE, LIST, DEL, ON-GOTO und ON-GOSUB. Bei Verwendung undefinierter Zeilennummern wird die entsprechende negative Zahl eingesetzt (beispielsweise GOTO-100). Bei berechneten Sprüngen (GOTO VAR, GOSUB 100+10★A, RESTORE A★10+1000) versagt RENUM. Wenn hinter dem Befehl eine Zahl folgt »GOTO 1000+A★10</br>
lie vird die Zahl wie bei einem normalen Befehl »GOTO 1000
lie behandelt. Der nachfolgende Programmteil der Zeile bleibt jedoch unverändert. Wenn keine Zahl, sondern ein Variablenname oder eine Klammer folgt, wird der Befehl überhaupt nicht verändert.

#### DUMP

**DUMP** filespec

Dieser Befehl erzeugt eine Liste der verwendeten Variablen. Wie bei LIST kann die Ausgabe auch auf einem Drucker erfolgen (DUMP "P:").

Ein Beispiel:

A = 100 numerische Variable

B( 10,1 Array, DIM B(9) oder DIM B(9,0)

C( 0,0 undimensioniertes Array. Bei Arrays werden die beiden möglichen Dimensionen stets um

eins erhöht angezeigt.

D( 10,10 DIM D(9,9)

E\$ 10,20 String, LEN=10, DIM E\$(20)
F\$ 0,0 nicht dimensionierter String
G\$ 0,10 DIM G\$(10), LEN(G\$)=0

H PROC 100 PROC H in Zeile 100 I # 120 Marke I in Zeile 120

J? Undefinierte Marke oder PROC

Die Ausgabe der Variablen/Marken erfolgt in der Reihenfolge, wie sie in der Variablentabelle gespeichert sind.

TRACE

TRACE +

Schaltet den TRACE-Modus ein. Das heißt, die Nummer jeder ausgeführten Zeile wird in eckigen Klammern auf dem Bildschirm ausgedruckt.

TRACE -

Hebt TRACE wieder auf. Der TRACE-Modus wird außerdem aufgehoben, sobald eine Fehlermeldung auftritt. Die Zeilennummern eines PROC oder # werden bei Aufruf mit EXEC oder GO # nicht ausgegeben.

**★B** 

**+B+** 

Nach diesem Befehl wird das Drücken der BREAK-Taste wie jeder andere Fehler behandelt. Eine Programmunterbrechung läßt sich mit TRAP abfangen und so vor einer versehentlichen Unterbrechung schützen.

\*B-

Hebt den oben erwähnten Modus wieder auf. Bei RUN wird > ★ B - « automatisch ausgeführt.

#### Befehle

Erläuterung:

<=> steht für: entspricht im normalen Basic

**DPOKE** adr,word

Doppel-Byte-POKE <=> POKE adr,word-256 \* INT(wort/256):POKE adr+1,INT(wort/256)

MOVE source, dest, count

Blocktransfer <=> FOR I=0 TO COUNT-1:POKE dest+I,PEEK(source+I):NEXT I

Mit »MOVE 57344,NEUCHARSET,1024« läßt sich beispielsweise der Zeichensatz kopieren.

MOVE source, dest, count

Blocktransfer, zuerst wird das letzte Byte verschoben. Dies ermöglicht, einen Speicherbereich zu einer höheren Adresse zu verschieben, ohne daß es bei Überlappungen (wenn source+count> dest) zu Zerstörungen kommt. <=> FOR !=COUNT-1 TO 0 STEP -1:POKE dest+I,PEEK (source+I):NEXT I. MOVE kann auch benutzt werden, um einen Speicherbereich zu füllen. Beispiel: »POKE DPEEK(88),128:MOVE DPEEK(88),DPEEK(88)+1,959«. Dies schreibt den Textbildschirm mit dem Bildschirmcode für das inverse Leerzeichen voll. Obwohl hier viele, eigentlich unnötige Ladevorgänge stattfinden, ist dies viel schneller als eine Basic-Schleife.

BPUT #n.adr.len

Blockschreiben <=> FOR I=0 TO ien-1:PUT #n,PEEK (adr+I):NEXT I

BGET #n,adr,len

Blocklesen <=> FOR I=0 TO len-1:GET #n,A:POKE adr+1.A:NEXT I

Mit diesen Befehlen kann man Speicherbereiche mit maximaler Geschwindigkeit speichern und laden. Beispiel: OPEN #1,8,0,"D:BILD.PIC":BPUT #1,DPEEK(88),7680: CLOSE #1

OPEN #1,4,0,"D:BILD:PIC":BGET #1,DPEEK(88),7680: CLOSE #1

Speichern oder Laden eines Grafik-8-(oder 9,10,11,15)-Bildes auf Diskette. Achtung: Die Zahl 7680 muß je nach Grafikmodus geändert werden (sonst werden nicht mehr zum Bildspeicher gehörende Speicherbereiche mitgelesen oder -geschrieben)

%PUT

Mit diesem Befehl lassen sich Zahlen schneller und kompakter auf Diskette oder in einer RAM-Disk speichern. Dabei werden immer sechs Byte gepeichert.

%GET

Zum Lesen von Zahlen, die mit %PUT auf ein Speichermedium gespeichert wurden. Ein Beispiel für die Anwendung von %PUT und %GET finden Sie auf Seite 70.

Kurzschreibweise, schneller und übersichtlicher als: >POSITION x,y:XIO 18, #6,0,0,"S."FCOLOR n<

Wählen der Farbe für FILLTO. In Standard-Basic heißt dieser Befehl »POKE 765,n«.

CLS

CLS #6

Bildschirmlöschen. CLS <=> A=PEEK(766):POKE 766,0:POSITION 0,0:? (#6;)CHR\$(125);:POKE 766,A

<=> ?CHR\$(n); beispielsweise »PUT 253« für »? CHR\$ (253);«. Bei PUT, GET, INPUT... ist auich die Angabe »#0« möglich. Die Benutzung von »IOCB 0« (mit »CLOSE« oder »OPEN #0«) verhindert allerdings das ordnungsgemäße Arbeiten des Bildschirm-Editors, gegebenenfalls SYSTEM-RESET drücken. Fehlt bei PUT die #-Angabe, so wird automatisch »IOCB #0« benutzt.

**GET KEY** 

<=> OPEN #7,4,0,"K:":GET #7,KEY:CLOSE #7. Wartet auf Tastendruck, und weist der Variablen KEY den ATASCII-Wert der gedrückten Taste zu (der Variablenname ist frei wählbar).

DKM

Beim DIM-Befehl werden Arrays und Strings automatisch gelöscht, also auf Null gesetzt. DIM A(100) <=> DIM A(100):FOR I=0 TO 100:A(I)=0:NEXT I

INPUT "text",var,var...

INPUT "text";var,var...

INPUT ähnelt jetzt sehr stark dem entsprechenden INPUT-Befehl unter Microsoft-Basic. Ein Text nach INPUT erspart die sonst nötigen PRINT-Befehle. Folgt nach dem Text ein Semikolon statt eines Kommas, so wird zusätzlich ein? ausgegeben. Mit »INPUT ""; A« erreicht man ein Input ohne das manchmal störende »?«.

TEXT x,y,sexp

Schreibt einen Text in ein Grafik-Bild. »x,y« stellt die Position der oberen linken Ecke des ersten Zeichens des String-Ausdrucks (wird in Bildpunkten gezählt) dar. Beispiel: »GRA-PHICS 8:TEXT 50,90,"Turbo-Basic":TEXT 70,95,1000«. Im Gegensatz zum normalen PRINT-Befehl darf nach TEXT nur ein Ausdruck erfolgen (keine Liste mit Komma oder Semikolon). Außerdem werden Texte am Zeilenende abgebrochen, es gibt auch kein Scrolling.

CIRCLE x0,y0,r

CIRCLE x0,y0,xr,yr

Zeichnet einen Kreis um den Punkt x0,y0 mit dem Radius r. Bei Angabe zweier (unterschiedlicher) Radien für die x- und y-Richtung entstehen Ellipsen.

PAINT x,y

Füllt eine geschlossene Figur, beispielsweise einen Kreis, mit der mit COLOR gewählten Farbe. Dieser Befehl kann fast jede beliebige Figur mit einer bestimmten Farbe füllen. Da hierbei eine für den 6502 angepaßte, rekursive Funktion benutzt wird, ist der Komplexität der auszufüllenden Figur durch den freien Speicherplatz eine Grenze gesetzt. Im Extremfall würde ein Arbeitsbereich von etwa 90 KByte RAM benötigt. Auch einfache und kleinere Figuren belegen vorübergehend einige hundert Byte. Sollte der Speicher übertaufen (FRE(0) zu klein), so gibt es die Fehlermeldung ERROR - 2 MEM«. Die PAINT- und die TEXT-Routine benutzen eigene, schnelle Plot-Routinen, da hierbei viele nebeneinander liegende Punkte relativ einfach berechnet werden können. Hiervon machen die Routinen im Atari-Betriebssystem leider keinen Gebrauch.

TIME\$= siehe unten

PAUSE n

Unterbricht die Programmausführung für n/50-Sekunden. PAUSE ersetzt das ungenaue und speicherplatzfressende Timing mit leeren FOR-NEXT-Schleifen. In Atari-Basic wird oft die Potenzierung (A=1^1) für eine kleine Verzögerung benutzt. Da Turbo-Basic XL bei der Potenzierung besonders schnelt ist, sollte dafür beispielsweise »PAUSE 9« benutzt werden.

FILLTO X,Y

DSOUND voice, freq, dis, vol

Ähnlich dem normalen SOUND-Befehl. Der Atari-Computer kann zwei normale Stimmen zu einer zusammenfassen. Die Frequenzauflösung beträgt dann 16 Bit (0..65535) statt 8 Bit (0..255). Die resultierende Frequenz berechnet sich (in Hertz) zu: 1789790/(2★freq+14) statt sonst 63921/(2★freq+2). Diese Werte stammen aus einer amerikanischen Veröffentlichung und unterscheiden sich eventuell etwas von den Werten bei den deutschen Atari-Versionen, da sie von der Videofrequenz abgeleitet sind, die In den USA einen anderen Wert besitzt (NTSC-System in den USA statt der europäischen PAL-Norm).

SOUND DSOUND

Kurzform für: FOR I=0 TO 3:SOUND I,0,0,0:NEXT I

CLOSE

Kurzform für: FOR I=1 TO 7:CLOSE #I:NEXT I

#### Funktionen

DPEEK(adr)

Doppel-Byte-PEEK <=> PEEK(adr)+256 ★ PEEK (adr+1)

INKEY\$

Spezialvariable. Wenn eine Taste gedrückt wird, enthält INKEY\$ das entsprechende Zeichen. Wenn nicht, enthält sie einen Leerstring (""). So läßt sich ein Tastendruck verarbeiten, ohne den Programmablauf zu unterbrechen.

INSTR(A\$,B\$) INSTR(A\$,B\$,I)

Sucht einen String B\$ in einem (längeren) String A\$. Wenn gefunden, wird die Position von B\$ in A\$ zurückgeliefert, sonst eine Null. »i« stellt den Index (Position) dar, ab dem die Suche beginnen soll.

UINSTR(A\$,B\$) UINSTR(A\$,B\$,I)

Ähnlich INSTR. Die Bits 7 und 5 der einzelnen Zeichen werden nicht beachtet. Bei der Suche nach »MODEM« läßt sich so auch »Modem«, »MoDeM« oder der entsprechende Inverse Text finden (UINSTR steht für UppercaseINSTR). Als Nebeneffekt werden bei der Suche nach Zahlen oder Satzzeichen auch die Spezial-Zeichen gefunden ("!"=CTRL-A,"0"=CTRL-P etc.).

ERR

Kurzform für »PEEK(195)« zur Ermittlung des Fehlercodes. ERL

Kurzform für »PEEK(186)+256 ★PEEK(187)« oder »DPEEK (186)« zur Ermittlung der Zeile, in der ein Fehler aufgetreten ist. ERR und ERL sollten in TRAP-Routinen verwendet werden.

TIME

Spezialvariable, enthält die Zeit (vom internen Timer RTCLOCK des Atari-Computers, ausschlaggebend sind die Speicherzellen 18 bis 20) in 1/50-Sekunden.

TIMES

Spezialvariable, enthält die Zeit als sechsstelligen String im Format hhmmss (hh=Stunde 00 bis 23, mm=Minute 00 bis 59, ss=Sekunde 00 bis 59).

TIME\$=

Zum Stellen der Uhr. »TIME\$="151520" « stellt die Uhr auf 15 Uhr, 15 Minuten und 20 Sekunden. Die Variable TIME läßt sich nicht direkt verändern. Statt dessen wird entweder »TIME\$=« verwendet oder durch entsprechendes POKEn in die Speicherstellen 18 bis 20. Die Uhr geht nicht ganz genau, da dle Frequenz, mit der im Atari die Fernsehbilder erzeugt werden, nicht genau 50 Hz beträgt. TIME\$ wird von dieser Frequenz abgeleitet.

FRAC(exp)

Diese Funktion ermittelt den Nachkommaanteil einer Zahl.

»FRAC(exp)« ist nicht immer gleich »exp-INT(exp)«, da INT die nächstkleinere Zahl ermittelt. So ergibt »? INT(-0.3)«-1, »? FRAC(-0.3)« ergibt -0.3.

TRUNC(exp)

Diese Funktion ermittelt den ganzzahligen Anteil einer Zahl. Dies ist die zu FRAC komplementäre Funktion. »? TRUNC(-0.3)« ergibt 0.

RND

RND (irgendwas) kann in Turbo-Basic XL abgekürzt werden, indem man die Klammern wegläßt. Es wird also RND statt RND(0) verwendet (speicherplatzsparend).

RAND(n)

Dies ist die Kurzschreibweise für »TRUNC(RND(0) \*\* n) « und erzeugt eine ganze Zufallszahl zwischen einschließlich 0 und (ausschließlich) n.

HEX\$(exp)

Ähnlich STR\$. HEX\$ wandelt die Integerzahl exp (0<=exp<=65535) in einen sedezimalen (hexadezimalen) String um. Wenn exp kleiner als 256 ist, ergibt dies einen zweistelligen String, sonst einen vierstelligen.

DEC(sexp)

Ähnlich VAL, Umkehrung zu HEX\$. Der String sexp wird in eine dezimale Integerzahl gewandelt. Wenn sexp mehr als vier gültige sedezimale Ziffern enthält, so gelten nur die letzten vier.

\$8888

Sedezimalzahl im Programm. Belspiel: FOR I=\$0600 TO \$067F:READ A:POKE I,A:NEXT I

FOR I=1536 TO 1663: READ A: POKE I, A: NEXT I

& Binäres AND

! Binäres OR

**EXOR** Binăres exclusiv OR

Diese drei Operatoren arbeiten mit 16-Bit-Integern. Also mit Zahlen zwischen 0 und 65535 und nicht mit Boolschen Größen (1 oder 0) wie die Operatoren AND, OR und NOT.

DIV

Division ohne Rest: a DIV b <=> TRUNC(a/b)

MOD

Bestimmung des Divisionsrestes (Modulo): a MOD b <=> a-b\*TRUNC(a/b)

%0 %1 %2 %3

Die Zahlen O bis 3 sind als Konstanten definiert. Die Verwendung einer Zahl (auch \$aaaa) im Programm kostet jedesmal 7 Byte, die Verwendung einer Variablen, unabhängig von der Länge des Namens, jedoch nur 1 oder 2 Byte. Auch die Verwendung von %0 bis %3 belegt ebenfalls jeweils 1 Byte, aber keinen Eintrag in der auf 256 Variablen begrenzten, Variablentabelle.

In Strings, oder in zwischen Anführungszeichen stehendem Text, ist es jetzt möglich, »"« (entspricht CHR\$(34)) durch doppelte Anführungszeichen ("") einzufügen. Beispiel: »?"TEST" "TEXT"« erzeugt den Ausdruck: »TEST"TEXT«. In Atari-Basic muß man dafür schreiben: »? "TEST";CHR\$(34); "TEXT"«. Bei Zuweisungen (A\$="TEST"TEXT") ist das (in Atari-Basic) noch etwas umständlicher.

Turbo-Basic XL wandelt bei der Eingabe von Programmzeilen automatisch kleine in große und inverse in normale Buchstaben um. Dies gilt natürlich nicht zwischen Anführungszeichen (") sowie nach REM oder DATA-Befehlen. Somit lassen
sich Programme auch mit Kleinbuchstaben eingeben, ohne
ständig zwischen Groß- und Kleinbuchstaben (mit der CAPSTaste) hin- und herzuschalten, oder ständig SHIFT drücken
zu müssen.

In Variablen- und Prozedurnamen ist außer den Buchstaben und Ziffern auch das Unterstreichungszeichen »\_\_« (SHIFT -) zugelassen. So sind endlich Namen wie MAX\_LEN oder PROC SORT\_KUNDEN erlaubt.

### Diskettenbefehle

DIR "D1: ★. ★"

Ausgabe des Inhaltsverzeichnisses einer Diskette auf dem Bildschirm. »DIR"D1: ★. ★ "« listet alle Files von Diskette 1. »D2:A★.★ « alle Files von Diskette 2, deren Name mit A beginnt. »D1: ★. ★ « kann entfallen, dann wird automatisch »D:★.★« eingesetzt (ähnlich wie im DOS-Menü mit »A RETURN RETURN«). Bei diesem Befehl sind Stringvariablen. wie auch bei den folgenden Befehlen, zulässig.

RENAME "D:OLD, NEW"

Benennt das File OLD in NEW um. Entspricht: »XIO 32, #7,0,0,"D2:OLD,NEW"« oder der DOS-Menü Auswahl E.

Löscht ein File (entspricht »XIO 33« oder der DOS-Menü Auswahl D)

LOCK "D:FILE"

Schützt ein File gegen Überschreiben (entspricht »XIO 35« oder der DOS-Menü Auswahl F)

UNLOCK "D:FILE"

Hebt den Schreibschutz wieder auf (entspricht »XIO 36« oder der DOS-Menü Auswahl G).

BLOAD "D:FILE.OBJ"

Lädt ein binäres File (entspricht der DOS-Menü Auswahl L mit »/N« nach dem Filenamen)

BRUN "D:FILE.OBJ"

Lädt ein binäres File und startet es, wenn eine RUN-Adresse im File enthalten ist (entspricht der DOS-Menü Auswahl L, ohne angehängtes »/N«)

Damit sind die zusätzlichen Befehle und Funktionen von Turbo-Basic XL erklärt. Selbstverständlich stehen darüber hinaus noch alle anderen Befehle, die man vom normalen Basic her kennt, zur Verfügung. Mit Turbo-Basic XI. wird die Programmierung auf dem Atari 800XL/130XE zur wahren

Freude. Befehle, die man sich schon lange gewünscht hat, stehen jetzt endlich zur Verfügung.

Autostart

Mit Turbo-Basic XL ist es möglich, nach dem Einschalten des Computers ein Basic-Programm automatisch zu laden und zu starten. Dazu benennen Sie Ihr Basic-Programm einfach in AUTORUN.BAS um. Vergessen Sie nicht. Das eigentliche Turbo-Basic muß dann mit dem Namen AUTORUN.SYS auf Diskette vorliegen. (Frank Ostrowski/wb)

# Turbo-Basic XL-Programme gesucht

Sicher werden Sie, so wie die Redaktion, sehr schnell von den Vorzügen von Turbo-Basic XL überzeugt sein. Deshalb an dieser Stelle ein Aufruf an alle Atari-Besitzer: Schreiben Sie doch Ihre Programme zukünftig in Turbo-Basic XL, und schicken Sie diese an die Redaktion. Schließlich wollen wir Ihnen nicht nur den Interpreter bieten, sondern Sie auch zukünftig mit Programmen in Turbo-Basic XL versorgen, die die größeren Fähigkeiten dieser Sprache voll ausnutzen.

Gefragt sind alle Arten Programme, Spiele, Anwendungen und Tips und Tricks. Versuchen Sie aber bitte, die Programme möglichst kurz zu halten. Ein Listing mit ein bis zwei Seiten Länge hat bessere Chancen, in Happy-Computer veröffentlicht zu werden, als längere (dies gilt übrigens auch für Programme in normalem Atari-Basic).

Schicken Sie Ihre Turbo-Basic XL und Atari-Basic-Pro-

gramme an:

Redaktion Happy-Computer z.Hd. Werner Breuer Hans-Pinsel-Straße 2 8013 Haar

0000:FF FF E7 82 E8 82 29 0148:80 01 D3 20 00 00 A9 FE<74> 0008:00 5E 87 5E 00 18 18 FF<E5> 0280:D8 6C D8 A3 24 38 D8 A2<44> \$148:80 81 03 A5 88 A4 81 80<98> 9919:30 30 66 42 98 88 88 86<14> @289:C7 94 C6 90 C6 89 C6 90<76> 9150:E7 02 8C E0 02 4C 5D E6<36> 8299:C6 87 C6 34 DA 63 DA 6618166 7E 76 66 99 00 18 3C<7C> 59(FE) 8158:00 22 C9 22 A3 24 A3 24<4D> 8298:25 88 D9 A6 D9 0020166 7E 66 66 00 3E 72 D9 DB<AC> 73 33(20) 8160:08 CB 28 C2 LD F1 B4 C2(93) 82A8:D9 55 DB 4C DB 89 D9 0028:3E 38 38 78 00 7C 43(7R) 37 33(64) 9168:00 E4 70 F5 BF F3 88 F4(95) @2A8:DB 84 DA 93 D9 0030:3E 30 30 78 00 00 78 DB 5E(49) 68 ASKEDS 8178:90 FF 90 FF 9A FF FC F5(D9) 8298: DB 67 DB 79 DB CD D8 0039:36 3C 39 63 00 01 01 0F<D4> BACSE'S 8178:11 25 E8 F5 A0 C8 40 C5(98) 0288: DA FF DA DE D9 D9 E2 0040:03 03 F6 E4 06 86 E6く6日> 80 F8<7C> 0180:58 F5 57 DC A0 CB AA F5KECS 8208:09 EA D9 1A 25 69 DC 0048:C9 C0 60 20 00 01 03 06<16> 77<65> 8188:65 E6 CC C3 E1 C2 43 C3(C4) 02C8:DC BS DD 4A DA 95 DC. 2A<83> 0050:06 86 83 81 88 E1 33 86<35> 8198: 8C C4 18 C4 2F C4 CB C3<A3> 8208: DA C2 D9 E4 DA C1 DA FC<4D> 2058:04 06 33 E1 00 C7 33(4A) @198:04 F6 44 25 F2 DC SA DC<66> 8208:FA C4 FA D8 DC A3 24 B6(RF) 0868:36 36 66 CE 80 0E 9C 9C<7F> 8148:75 CA 62 CA 18 FB 32 F5(B2) 02E01DA AD DA E7 DA 9E 0068:F6 66 66 67 00 7E 33 DB AC(89) 33(A3) 81A8: 98 F5 D3 F7 F2 DC 67 C4<58> 02E8: D8 A6 D8 A3 D8 A3 24 B5<CE> 0070:3E 30 30 78 00 67 66 8188:54 C4 53 C2 74 C2 18 C2(F7) 66<58> 02F0:DD E4 DC E9 DC 00 00 20<45> 9078:66 66 66 3C 80 FF 31<A4> 31 @188:FE 24 38 C2 AE C1 95 C4<85> 82F8:88 88 88 88 88 88 80 80 82 88<FC> 0080:31 31 31 78 60 FB 81 B1<A1> @1C@: D9 C1 97 DD 8A C3 92 C3(49) 6386:88 86 28 28 28 28 28 28 28(ED) 00881F1 81 81 FB 88 FB 8C 8C(C4) 81C8:00 E4 DA FB 32 25 00 D8<94> 6386:2C 28 22 22 28 1E 1A 1C<9E> 0096:F8 100 98 CC 88 21 93 21<33> 0100:00 08 FF F6 1E F4 98 F7(FA) 8318:32 84 32 32 38 38 38 38(6A) 9098: A2 00 BD 00 E0 9D 00 5C<52> @1D8:84 F7 B9 F7 BD F5 98 F5(49) 0318:30 30 2E 2E 32 32 32 32(9B) 00A0: BD 20 E1 9D ma. 50 BD 88(54) 81E8: DA C5 DD C5 35 C2 1E F4<E9> 8328:32 84 32 32 32 32 32 32(58) 08A8:E3 90 00 5F E8 D8 EB A9(34) 81E8:88 F8 52 F8 A8 C5 C6 C5<7A> 6328:32 32 32 32 32 32 32 32(2D) 00B0:10 BD C6 82 A9 50 BD C5<49> 01F0:09 C5 C3 C5 CC C5 A2 DC<CF> 0338:32 32 32 32 32 32 32 32(15) 0088:02 8D F0 02 8D F4 82 A9<E4> 01FB: 74 FB B3 F8 82 F8 64 F8(FC) 0338:32 32 32 2A 2A 32 32 2A(75) 00C0:45 8D 44 03 A9 21 8D 45(67) 8288: 2E C2 92 F3 A3 24 E4 33<AD> @340:32 32 29 32 32 32 28 00(BE) 00C8:03 A9 4F 8D 48 03 8E 49(11) 0208:00 CO 40 CO 58 FF 37 FC<49> #348:32 32 32 32 32 32 32 00<1A> 00000:03 A7 08 8D 42 03 20 56<93) 8218174 35 77 35 AF F8 A3 24<7C> 0350:32 32 32 00 06 00 00 00<22> 00008:E4 8E F0 02 40 7D 9B 7F<35> 0218:99 F5 D0 FD 3F FF D9 C1(59) 9358: 80 66 70 68 70 60 60 60 00E0: 7F 28 88 81 82 83 84 85<61> 00<5E> 0220:80 DE 18 C6 18 C6 00 23(52) 0368:28 28 28 28 28 28 2C 00E8106 07 9B 28<64> 7F 7F 08 09 0A<31> 8228:15 23 EA 28 24 23 4C 2B(2C) 0369:22 22 28 10 1A 1C 02 30<13> 00F0:08 0C 0D 0E 0F 10 78 78<86> 0230:E5 44 3A 2A 2E 2A 98 53<33> 8370:01 81 38 38 38 38 38 30<02> 00F8:7F 20 20 20 20 54 55 52(50) 8238:3A 9B 43 3A 9B 58 3A 9B(31) 0378:2D 2D 02 02 02 02 02 03<3F> 0100:42 4F 2D 42 41 53 49 43<D9> 8240:24 23 2E 23 AD 81 D3 29<2F> 9380:02 02 02 02 02 02 02 02 62<86> 0108:20 58 4C 20 31 2E 35 98<01> 9248:FC 99 82 8D 81 D3 68 3AKE3> 2388:02 **2**2 02 02 02 02 02 02(BE) 0110:7F 20 28 63 29 20 31 39<6A> 0250:23 28 36 45 DB 4E DB 65(17) 0390:02 02 02 02 02 02 02 02(96) 0118:38 35 20 46 72 61 6E AB< OR> 0258:00 55 00 5C 00 6C 00 9F<00> 0398:02 ZA 2A 82 32 2A 82 82<C7> 0120:28 4F 73 74 72 6F 77 73<2A> 0240:DB 26 DB 14 DB 1D DB 2F<A9> \$3A8:28 \$2 32 32 28 88 32 82<2A> 0128:68 69 98 98 E2 02 E3 02<16> 0268:D8 95 D8 84 D8 73 D8 A3<D4> 93A8+82 32 32 32 32 22 22 32(4D) 0130:00 21 80 20 A1 20 A7 B0(C3) 8278:24 89 C6 4C C6 1C C8 0380:32 EE 01 D3 20 56 E4 CE<08> 0138:A0 20 85 0C 84 0D A9 FF(0C) 0276:08 4E D8 65 D8 55 D8 5C(6D) 0388:01 03 C0 00 60 68 AA CE<AF>

Listing »Turbo-Basic XL«. Bitte unbedingt mit dem Eingabe-Programm AMPEL von Seite 87 eintippen



		_			_		
Ø3CØ: 01	- 03	68	40	2C	ØF	D4	10<82>
03CB: 03	6C	20	02	48			
03D0:24	48	A9	94	48		BD	
03DB: 01	48	DB	48	8A			
03E0: EE	01	D3	8D	ØF	D4	6C	22<03>
<b>0</b> 3E8:02	48	A9	24	48	A9	A6	48 <b3></b3>
03F0:08	EE	21	<b>D</b> 3	6C	16	02	EE<71>
Ø3F8:01		20	EB				
	D3			24	CE	21	D3 <e4></e4>
6460:40	BD	47	03	48	BD	46	@3<43>
0408:48	78	AB	SD	69	EE	@1	D3<53>
8418:28	80	28	CE	61	D3	60	20<39>
0418:34							
	C5	EE	01	D3	AD	8E	20 <c7></c7>
0420:AC	- epr	20	85	ØC.	84	ØD	&C<73>
0428: QA	80	20	34	C5	EE	01	D3<2F>
0430:40	71	E4	20	16		EE	Ø1<5A>
Ø438: D3		@1	B1	D4		88	B1 <d4></d4>
Ø440: D4	CE	01	D3	85	D4	86	D5<50>
0448:4C	7D	D9	26	ØC	E5	EE	01<83>
0450:03	AØ	00	91	78	A5	D5	CB <d4></d4>
9458: DØ		4C	26	E5		ØC.	E5 <bb></bb>
0460:DØ	FB	EE	01	D.3	71	78	AD <d7></d7>
0468:01	D3	29	FC	09	02	BD	@1 <f6></f6>
0470:D3	60	20	62	25	28	DC	2B <a@></a@>
Ø478:40	3A	E2	A9	23	49	48	AS <dc></dc>
Ø48Ø: BØ	85	CP	20	16	E5	CA	C6<72>
0488:30	29	A5	<b>D4</b>	48	A5	05	48 <a3></a3>
0490:4C	6A	25	EE	01	D3	A5	BØ(ED)
0498:48		D4					
	60		99	4C	7B	F6	A9<3F>
0400:00	84	A4	85	A5	18	A5	90 <c5></c5>
Ø4AB: 65	A4	AB	A5	91	65	A5	CD <cb></cb>
04B0:E6	02	90	09	DB	E6	CC	E5<61>
0488102	70	02	DØ		28		
	_			DF		A5	90<04>
04C0:F3	99	85	A2	A5	91	F5	01<70>
Ø4C8:85	A3	18	B5	20	85	97	85<10>
04D0:99	65	A4	85	78	B5	01	B5<7D>
04D8:98	85	9A	45	A5	85	90	B5<8Ø>
04E0:00	65	A4	95	30	85	01	65(29)
Ø4E8: A5	95	21	E8	E8	EØ	92	46<88>
04F01EE	85	ØF	A5	70	85	ØE.	EEKE9>
@4FB: 01	D3	86	A3	18	BA	65	9A <a2></a2>
0500:85	9A	18	BA	65	90	85	90<57>
0508:E8	A4	AZ	FΦ	25	68	B1	99 <a4></a4>
0510:91	98	96	DØ	F8	FØ	1B	C6<15>
051819A	Ca	90	88	B1	99	91	98(A1)
0520:08	B1	99	91				
				7B	88	B1	99 <a4></a4>
0528:91	99	88	B1	99	91	9B	98 <a2></a2>
Ø530:DØ	E9	CA	DØ	E2	CE	91	D3 <ad></ad>
<b>0538:68</b>	A9	00	84	84	85	A5	38<87>
0540: A5		F5					
	90		96	85	A2	A5	91<04>
0548:F5	01	85	A3	38	85	60	85 <f9></f9>
D550:99	E5	A4	85	9B	B5	01	85\DB>
0558: 9A	E5	A5	85	90	38	85	00<8C>
0560:E5	A4				-		
		95	88	85	21	E5	A5<9A>
Ø548195	01	ĒΒ	EB	EØ	92	90	ED <d1></d1>
Ø57Ø185	ØF	A5	70	85	ØE	EE	Ø1<37>
Ø578: D3	AZ	00	A6	A3	FØ	1D	B1<7E>
0580:99	91	79	CB			91	
				BI	99		4B<23>
<b>6</b> 298: CB	B1	99	91	98	CB	81	99<61>
Ø59Ø191	9B	CB	DØ	EA	E6	9A	E6<08>
Ø598:90	CA	DØ	E3	A6	A2	FØ	図8<4B>
05A01B1	99	91	9B	CB	CA	DØ	FB(42>
Ø5AB: CE	01	D3	60	18	4C	90	2B<8C>
0580:18	60	3B	60	20	90	2E	A5<40>
05BB: D4	FØ	F5	A5	EØ	FØ	ED	45<6F>
0500: D4	29	88	85	EE	A5	EØ	29<21>
05C8: 7F	85	EØ	A5	D4		7F	
					29		3B<33>
05D0:E9	40	38	65	EØ	30	DB	05<13>
0508:EE	AB	20	9E	ZA	85	DA	85<3D>
ØSEØ: DB	85	DC	85	DD	85	DE	85 <cb></cb>
05E8: DF	84	<b>D4</b>	AD	27	46	E5	90<7A>
05F0:30	18						
		A5	DE	79	FZ	22	85<67>
05F8: DE	A5	DD	79	EΑ	22	95	DD<8B>
0600: A5	DC	79	E2	22	85	DC	A5<88>
0608: DB	79	DA	22	85	DB	A5	DA <fø></fø>
0610:79							
	D2	22	85	DA	A5	D9	79<7A>
0618: CA	22	85	D9	88	10	CE	30<05>
0620:05	FØ	03	88	10	C7	AB	87<69>
0629:46	E4	90	30	18	A5	DD	79 <cb></cb>
8630 x F2	22	85	DD	A5	DC	79	EA <bd></bd>
0638:22	85	DC	A5	DB	79	E2	22<2B>
0640:05	DB	A5	DA	79	DA	22	85×80>
Ø648: DA	A5	D9	79	D2	22	85	D9<54>
		79					
0650:A5	08		CA	22	85	DB	88 <cf></cf>
Ø658: 19	CE	30	<b>Ø</b> 5	FØ	03	88	10<00>
0660:C7	AØ	07	46	E3	70	30	18<0E>
Ø668: A5	DC	79	F2	22	85	DC	A5<80>
Ø67@: DB	79	ĒΑ	22			A5	DA< 1A>
				85	DB		
Ø678:79	E2	22	85	DA	A5	D9	79<16>
0680: DA	22	85	D9	A5	DB	79	D2 <d1></d1>
Ø688:22	85	DB	A5	D7	79	CA	22<6A>
0690:85	D7	98	10	CE	30	25	FØ<30>
0698:03	88		C7	AØ	67	46	E2 <fc></fc>
06A0:90	30		A5	BB	79	F2	22 <eø></eø>
Ø6A8: 85	DB	A5	DΑ	79	EΑ	22	85<9F>

```
ØABØ: DA
          A5 D9 79
                    E2 22 85 D9<EF>
 MARS: A5 DR
             79
                DΔ
                    22
                        商等
                           DB
                              A5<515
MACH: 07
             D2
                    85 D7
                 22
                           AS.
                              DAK 5B3
 96CB: 79
          CA
                        88
             22
                 85
                           18
                    D6
                              DEK D30
Ø6DØ: 3Ø
          PM5
                    88
                               ARK RE
                        18
96DH: 07
          46
             F1
                 90
                    30
                           45
                              DAK1C)
06E0:79
          F2
             22
                 85
                    DA
                        A5
                           09
                               79<C2>
 DAEB: EA
          22
             85
                 D9
                        DS
                    A5
                              EZKF@>
                           79
06F0:22
          B5
             D8
                 A5
                    DZ
                        79
                           DA
                               22-A23
ØAF8: 85
         D7
             45
                 06
                    79
                        02
                           22
                              85<91>
0700: D6
          A5
             D5
                 79
                    CA
                        22
                           85
                              D5<99>
                              BB< 91>
                 30
                        EB
          10
             CE
                    05
                           03
871B:1B
         C7
             4C
                    2D
                        18
                           69
                               38<4F
                FØ
                              F@<B4>
271B: 60
         45
             FOL
                    FA
                        A5
                           D4
0720:F4
          45
             EB
                 29
                    AD.
                       B5.
                           FE
                              45(44)
0728:FB
         29
             7F
                 85
                    ÉØ
                        A5
                              29(84)
                           D4
0730: 7F
          38 E5
                EØ
                    18
                        69
                           40
                              30K 793
                    20
07.38: DE
          M5
             EE
                 AB
                        9E
                           ZA
                              85(2F)
0740:E6
         85 F7
                85
                    F8
                       85
                           F9
                              85(98)
Ø748: EA
         85
             EØ.
                           D4
                85
                    DA
                        B4
                              ARK 1D>
0750:00
         A5
             EØ
                    CA
                        22
                           DØ
                              28(87)
         E1
0758: AS
             D9
                D7
                    22
                        DØ
                           21
                              A5<60>
                              E3(06)
07A01F2
         D9
             DA
                22
                    DØ
                        18
                           A5
                DØ
0768: D9
             22
         E2
                    13
                        A55
                           F4
                              DRICACO
0770:EA
          22
             DIA
                    A5
                ØC.
                       ES
                           D9
                              F2<13>
0778:22
         DØ
             05
                AZ
                    00
                        4C
                           95
                              ZAKCAS
0780-9B
         24
             45
                E5
                    F9
                       F2
                           22
                              B5(RF)
078B: E5
         85
                       22
             F4
                F9
                    FO
                           85
                              F44455
0790: A5
             F9
                E2
                    22
         E3
                       85
                              A5<F1>
                           E3
0798:E2
          FQ
                    85
             DA
                22
                       E2
                           A5
                              E1<11)
                       A5
                              F9(6E)
M7AM: F9
         02
             22
                85
                    E1
                           EØ
D7AB: CA
         22
             85
                EØ
                    26
                       D5
                           CB
                              COKAGO
07B0:08
         DØ
                    99
                              D9<D5>
             9E
                AB
                       A5
                           E1
0788: CA
          22
             DØ
                28
                    A5
                       E2
                              D2< D90
07C0:22
         De
             21
                A5
                    E3
                       D9
                              22(92)
                           DA
27CB: DØ
         10
             A5
                F4
                    D9
                       E2
                           27
                              DØK SAD
07DØ:13
         A5
             E5
                D9
                           DØ
                    EA
                       22
                              @CKSE>
Ø7DB: A5
         E6
             D9
                F2
                    22
                       DØ
                              A2(E9)
                           05
97E0:01
         4C
             95
                26
                    94
                       2A
                           A5
                              E6KEC)
                           E5
07E8:F9
         É2
             22
                85
                    E6
                       A5
                              F9<11>
         22
07F0:EA
             85
                    A5
                       E4
                E5
                           E9
                              E2<E40
07F8:22
         85
             E4
                A5
                    E3
                       F9
                           DA
                              22(21)
             A5
0800:85 E3
                E2
                    F9 D2
                              85<26>
2808: E2
         45 F1
                F9
                    CA
                       22
                           95
                              E1KEB:
0810:26
                CØ
                    98 D8
         06
             CB
                           95
                              ARK RRD
         A5
                       22
0818:00
             E2
                D9
                    CA
                           DØ
                              28(99)
Ø820: A5
         E3
             09
                D2
                       DØ
                    22
                           21
                              ASKEB)
0828:F4 D9
             DA
                22
                    DØ
                       18
                           A5
                              E5<F3>
0830:D9
                DØ
         E2
             22
                    13
                       A5
                          FA
                              D943A3
         22
             DØ
Ø839: EA
                ØC.
                    AS
                       E7
                           09
                              F2(BC)
0840122 D0
             25
                A2
                    02
                       4C
                           95
                              2A<A4>
2848: 92
         2A
             AS.
                E.7
                    E9
                       F2
                           22
                              85(26)
            E6
0850: E7
         A5
                FP
                    EA
                       22
                           85
                              E6<50>
0050:A5 E5
             F9
                E2
                    22
                       85
                          E5
                              A5<C3>
3869:E4
             DA
                22
                    85
                       E4
                           A5
                              E3<FE>
08681F9 DZ
                85
             22
                    E3
                       A5
                           E2
                              F9<60>
0970:CA
         22
            85
                E2
                    26 D7
                           08
                              CB< 19>
0879:08 D0
             9E
                AØ
                    88
                       A5 E3
                              D9(RE)
2082:CA
         22 DØ
                28
                       E4
                    A5
                              D2<37>
                          D9
                A5
                    E5
Ø888122
         DØ
             21
                       D9
                           DA
                              22(50)
0970 DB
         1A A5
                E6
                    09
                       E2
                           22
                              DØ<64>
0890:13
         45 F7
                DQ
                    FΦ
                       27
                           DØ
                              BC < 495
SHARL AS
         EB
             D9
                    22
                       DØ
                F2
                              AZKAED
                           05
08A8:03 4C
             75
                2A
                    70
                       2A
                           A5
                              EB<BØ>
0890:F9
         F2
             22
                85
                    EB
                       A5
                          E7
                              F9(EQ)
                       E6
         22 85 E7
0888: EA
                    A5
                           E9
                              E2<85>
Ø8CØ: 22
         85 E6
                A5
                   E5
                          DA
                       89
                              22(@A)
00CB: 95
         E5
             A5
                E4
                       D2
                           22
                              85<FF>
08D0:E4
         95
                F9
                    CA
                              E3<F3>
            E3
                       22
                           85
ØBDS: 26
         DR
            CB
                CØ
                   Ø8
                       DØ
                           9F
                              AØ<433
08:0380
         A5
            E4
                D9
                   CA
                       22
                           DØ
                              28(02)
08E8: A5
         E5
            09
                DZ
                    22
                       DØ
                           21
                              A5<14>
                              E7<1E>
08F0:E6
         D9
             DA
                22
                    DØ
                       18
                           A5
M8F8: D9
         E2
            22
                Det
                   13
                       A5
                              D9(92)
                           EB
09001EA
         22
                   A5 E9
            DØ
                ØC
                          DP
                              FZ<BE>
         DØ
0908:22
            25
                A2
                   24
                       4C
                           95
                              2A<7E>
                   F9
                          22
8910:98
         28
            A5
                E9
                       F2
                              85< 78>
                              EB<1B>
0918:E9
         A5
            EB
                FQ
                   EΑ
                       22
                           85
0920:A5 E7
            F9
                E2
                   22
                       85
                          F7
                              A5<55>
         F9
                22
                   85
0928:E6
            DA
                       E6
                          A5
                              E5<BC>
0930:F9
         02
            22
                85
                   E5
                       A5
                          E4
                              F9<44>
                              DØ<6A>
0938: CA
         22
            85 E4
                   26
                       D9
                          CB
0940:08 D0
            9E
                A55
                   D5 D8
                          AA
                              ARK RES
0948:00
         A5
            E5
                D9
                   CA
                       22
                          DØ
                              28<5C>
09501A5
            D9
                D2
                   22
                       DØ
                          21
                              A5< 90 >
0958:E7
         D7
            DA
                22
                   DØ
                       10
                           45
                              E8(7F)
0960:D9 E2
            22
               DØ
                   13
                      A5
                          E9
                              D9<72>
0968:EA 22
            DØ
               ØC
                   A5
                          D9
                              F2<2B>
                      EA
0970122
         DØ
            65
                A2
                   05
                       4C
                          75
                              2A(EE)
69 7A - 9B
         2A
            A5 EA
                   F9 F2
                          22
                              85(E9)
8988: EA A5
            E9
               F9
                   EA
                       22
                          85
                              E9<67>
0788: A5 E8 F9 E2
                   22 B5 E8 A5<40>
0990:E7 F9
            DA
                22
                   85
                      E7
                              E6<98>
                          A5
```

09A0:CA 22 85 E5 26 DA CB C0<13> 36 89A81 08 DØ 9E 4C 84 2D D5<BE> MARNICS CO NO DO F9 FØ E4 F8KFC) 9988: 18 95 ES BD F9 22 65 F5<83> M9CA: RD EB フワ E4 BD A5 F1 22(FF) 09CB: 65 22 A5 E3 BD<AD> ASDA: ES 22 65 **E**3 SD E8 22 ASKBE > 09DB: E2 BD E1 22 65 E2 ßb EØ<23> 09E0:22 **A5** 8D 119 22 E1 65 E1KCC> @9E8:8D DB 22 49 00 80 D1 22<3C) 09F0:69 MA AD DR 22 BD(9D) DOFR: FA 77 70 FA 72 9D F5 22<2C> MAMOR BD EE 22 7D EE 22 9D FDCAAD 9A08: 22 22 E6 BD E6 7D 22 9D(AB) 0A10:ES BD DE 22 7D 22(79) 7D D6<48> 0618-9D DD 22 BD D6 22 9A28: 22 05 22 90 RD CF 22 7D(AE) DAZE: CF 9D CD 22 C7<CB> 22 CA D/A MAJO: AD FE 22 F5<E9> 4D F6 22 80 ØA38:22 AD FØ 22 6D EE 22 BD(9E) 0A40: ED 22 AD E8 22 6D EA 22(15) 22 6D DE<18> BA48: 8D E5 AD EØ 22 ØA50:22 BD DD 22 AD D8 22 ADKCC) 0A58: D6 22 8D D5 22(54) 22 AD DØ RAARLAD CE 22 BD CD 22 A2 Ø2<515 22 0A68: BD F3 22 7D F3 PD. F7(FF) 9D<87> ØA70:22 BD EB 22 7D EB 22 22 BAZB: EA BD E3 7D E3 22(06) MARM: 90 F2 22 Bn DB 22 7D DB<RED BA68: 22 9D DA 22 **AD** D322 70(29) ØA98: D3 22 9D D2 22 BD 22<FE> CB 0A9817D CB CA 22 9D 22 CA 10<96> MAAB: C7 A5 05 85 E1 **A5** D<sub>6</sub> 85(DE) BAAG: EZ AS D7 85 E3 **A5** DB 85(EC) 0AB0: E4 A5 D9 E5 85 A9 98 85(92) 85 BABE: D4 05 85 D6 85 85(C2) F4<AB> MACM: DB 85 D9 60 49 05 95 MACH: A9 BB 85 E3 68 EE. @1 D3<60> 0AD0:28 A1 DB CE @1 D3681 存在く570 E6 D3<4F> GADE: Ø1 D320 DB CE @1 BAFA: AR FF @1 D3 20 20 DB CEKFC) @AEB: @1 D3 60 **A4** F2 B1 F3 3B(EC) MAFO: E9 30 C9 ØA. 60 **D4 A4** A5<76) 0AFB: 05 85 20 90 2B FB 98(12) 0B00: F0 2E 40 48 46 48 85 F8<00> **BBBB: 98** 29 87 ΔR (A) 90 02 69<4A) QB10:07 46 FB 99 82 69 15 46<55> 0818:F8 90 02 FR 98<F7> 69 31 46 0B20:05 69 63 90 01 CB 46 F8<E7> Ø828:90 Ø6 69 27 Ca 61 C8<71> 70 08301 AA F7 FA 6F 46 F7 90 M7(FE) @B 18169 CO 92 55 CB 21 ČB. 46(B3) 0840:F7 98 69<56> 90 08 11 AA 69(E3) 0B48:05 AB RA 46 E7 90 68 46<88> 69<00> ØB50:23 AA 98 69 10 AB BA 08581F7 98 Ø8 69 47 98 AA **0860:20** AB BA 46 F7 **28** 69(38) 69 **ØBAB195** AA 98 40 46<F6) AB BA ØB70:F7 90 ØC: 69 91 AA **9**B 69(01) **0**978:81 AB. BA 98 02 46<3A> EA 05 0880:F7 98 ØE 69 83 78 69(E1) AA **20088163** AB D5 69 01 85 D5<89> A5 AAKEZ> 0890: RA 44 F7 98 ONE 69 67 **2**878: 78 27 69 AB A5 D5 Ø3<02> 49 A9<57> 08AØ: 85 D5 BA 84 Do D7 85 **PBAB: 42** 85 D4 4C BB 2D 60<5A> 0980: A5 EA 49 RIA. 85 EØ 45 E@<@70 **ØBBB: 29** 7F FØ A5 D4<47> F2 85 F7 ØBC0:29 7F 38 E5 F 7 BØ A5<E6> 33 ØBC8: D4 **A4** 04 EØ 85 EØ. 84 A5<93> 9800: D5 A4 E1 85 E1 84 ns A5154> ØBDB: D& **A4** E2 85 E2 84 D6 A5< D5> 08F0: 07 44 E3 E3 85 84 **D7** A5<77> 08EB: 08 A4 **E4** 85 **E4** 84 DB A5<32> E5 4C(FB) 0BF6: D9 A4 E5 85 84 09 20 **ØBF8:9D** AB F8 4B 88 FØ 36<8A> 0C00:88 FØ 22 69 FØ 88 DØ<85> 10 BCB8: A5 **A5** E1 85 E5 84 E4 84<09 0C10:E3 B4 EZ 4C 2D2D **A5** E2<05> ØC18:85 E5 A5 E1 85 E4 84 E3<7E> ØC20:84 E2 4C 2D 2D **A5** 85<92> Ë3 0C28:E5 A5 E2 65 E4 A5 E1 85(62) 0C30:E3 84 E2 2D 2D E4<E7> 4C A5 WC38:85 ES A5 F.3 85 F4 A55 F2く7F> QC40:85 A5 E1 65 E2 84 E1<89> E3 0C49: FB **A5 D4** EØ 30 18<9C> 0C50: A5 D9 65 E5 09 A5 08< 980 **B**5 DC58: 65 E4 85 DB AS **D7** 65 E3KDF3 0060:85 D7 A5 D6 65 E2 85 DAKE4> 0C68: A5 05 65 85 16(F9) E1 05 90 0070:A5 D8 85 D7 85 DB(E3) 0078:A5 D6 85 D7 A5 D5 85 D6<2E> 0000:A9 01 85 D5 EA D4 4C RB(7A) 0C88:20 38 A5 D9 E5 E5 85 D9<E6>

@9981F9 D2 22 85 E6 A5 E5 F9(B4)

@C98:A5	08	೯೮	E4	85	DB	A5	D7 <f4></f4>
0C98:E5	£3			A5	D6		
		85	D7			€5	E.2<ØA>
QCA0:85	DA	A5	D5	E5	E1	85	05<5B>
0CA8: B0	27	AS	D4	49	80	85	D4<7B>
@CR@:38	98	€5	09	85	D9	98	£5<92>
OCE8:08	85	D8	98	E5	D7	85	Q7 <b6></b6>
ØCCØ: 98	E 5	D6	85	D6	98	E5	D5<94>
0CC8: 85	05	4C	BB	2D	A6	DA	DØ <d1></d1>
0CD0:02	A2	86	80	04	D4	FØ	6D<99>
0CD8: 45	05	DØ	5C	88	A5	D6	DØ <ce></ce>
ØCE0:47	88	A5	D7	DØ	30	68	A5<3B>
20CE8: 08	D2	1B	88	A5	D9	DØ	<b>0</b> 8<70>
OCF 0:88	SA	FØ	51	85	D5	DØ	48<51>
QCF 8: 85	D5	86	06	49	88	85	D9<50>
9000:85	08	85	D7	FØ	32	85	D5 <f7></f7>
0008:A5	09	85				A9	
			90	88	D7		80<1A>
GD 10:85	09	85	DB	F	22	85	D5<37>
QD18:A5	DB	85	D6	A5	D9	85	D7<19>
ØD20:86	08	49	00	95	D9	FØ.	10(09)
0028:85	25	A5	D7	85	06	A5	D8<9C>
00 TO: 05	07	A5	09	85	DB	84	094EE>
@D38:84	D4	98	29	75	C9	71	BØ<56>
2040:05	C9	ØF.	PØ	03	20	90	2B<7A>
2D48: 18	60	86	FL	84	FD	85	EF (EØ)
@D50:20	62	2E	20	7C	2E	C&	EF<7C>
@D58:20	9E	26	BØ	37	A5	FC	69<16>
9D96:69	65	FC	90	02	E6	FD	20(08>
0D68: 7C	2E	20	90	2C	BØ.	25	E6<50>
	FØ						
0070:EF		21	AD	E.8	05	85	EØ< 18>
@D78: AD	E1	05	85	E.I.	AD	£2	05<10>
0000:85	E2	AD	ES	85	85	E'2	AD<4A>
2088:E4	05	85	E4	AD	E5	05	B5<20>
0090:E5	4C	3F	2£	60	AB.	05	B1<37>
2099:FC	85	E.5	88	B1	FC	85	E4<48>
0DA0:88	B1	FC	85	E3	88	B1	FC(87)
@DA8: 85	E2	88	81	FC	85	E1	88 <f9></f9>
ØDBØ: B1	FC	85	EØ	60	A5	D4	85< 93>
SD88: E0	AS	D5	B5	E1	A5	D6	B5 <e9></e9>
2DC0:E2	A5	D7	85	E3	A5	DB	85<50>
2DC8: E4	A5	D9	85	E5	68	A5	D4<69>
@DDØ: 80	EB	95	AS	05	8D	€1	<b>85</b> <83>
RDDB: A5	06	8D	E2	25	A5	D7	BD <a0></a0>
00F0:E3	05	A5	DB	BD	E4	05	A5< 76>
0DE8:09	SD	E5	05	60	A5	D4	8D <c1></c1>
ODF0: E6	05	A5	D5	8D	E7	05	A5<57>
BDF8:06	as	£8	85	A5	סס	8D	E9<92>
ØE ØØ: Ø5	A5	90	80	EA	<b>6</b> 5	A5	D9<21>
ME 08:80	EB	05	68	A2	89	4C	FF <f@></f@>
GE 10: 26	A2	86	4C	FF	2ξ	A2	ØE<63>
ØE 18: 50	EQ	65	85	04	000	E1	<b>65</b> <cb></cb>
ØE 20:85	95	BD	E2	105	85	D6	BD (BE)
	-						
QE 38: E2	Ø5	85	D7	BD	E4	<b>0</b> 5	B5 <f9></f9>
@E 30: DB	8D	E2	05	85	D9	68	A2<40>
ØE 38: ØC	40	25	2F	A2	06	CE	EB<622>
ØE.40: 95	85	68	BD	E.I	65	85	E1KED>
ME 48: BD	E2	<b>Ø</b> 5	85	E2	BD	E3	@5<31>
0650:85	£3	BD	€4	05	85	E4	BD< 20>
0E58:E5	05	85	E5	68	A2	85	EE (AØ)
			COL				
ØE 60:01		BD		DE			CAKA@>
	FB	CE	01	D2	58	9E	26(CC>
ØE 78: 80	EA	AP	88	85	F1	A5	04<7B>
ØE 78:85	FØ	29	7F	85	D4	C9	40<81>
ØE 80: 90	1E	DØ	DE	A5	D5	29	FB<9E>
@E88:4A	25	F1	40	40	65	F1	85 (34)
0E'90:F1	AS	D5	29	ØF	65	F1	85KEC>
ØE'98:F1	A9	66	85	D5	20	18:8	20(73)
						E.E	
BEAD: A9	ØA.	AZ	40	AØ	DE		01 <ed></ed>
@EA8: D3	20	31	26	CE	01	D3	20<77>
ØE90:98	26	A5	F1	FØ	ØF	40	18 <cf></cf>
SE BB: 65	D4	20	16	85	D4	46	F1<7G>
ØECØ: 98	62	20	98	32	86	FØ	90(2D)
DECE: DA	28	90	2E	20	Бb	2F	4C <de></de>
0ED8: 08	28	38	60	99	40	85	D4<19>
DEDB: AO	01	84	05	88	84	D6	84<64>
ØEE0: D7	84	DB	84	D9	60	86	FC <d9></d9>
ØEE8:84	FD	20	85	2E	20	7C	ZE <ed></ed>
QEF0:20	9D	20	20	04	2E	20	F3<55>
@EF8:2E	20	7C	ZE	22	97	2C	20<59>
0F00:23	2F	40	88	28	38	68	A9< 70>
@FØ8: Ø5	De	02	A9	88	85	FB	A5 <fe></fe>
ØF 18:04	30	FZ	FØ	FØ	ØA	49	88<1B)
ØF 18:85	FI	49	40	85	D4	A5	D5 <e8></e8>
ØF 20:29	FD	FØ	25	E6	F1	20	A3 <f3></f3>
@F28: 52	AZ	66	AØ	DF	E.E.	61	<b>D</b> 3<22>
@F30:20	CD	2F	20	04	2£	20	9B <ac></ac>
ØF 38: 26	A9	8A	A2	72	AB	DF	20(E5>
0F40:31	2E	CE	01	D3	20	23	2F<8B>
0f 48:28	9E	26	A9	3F	85	EØ	A9<91>
0F50:50	85	13	A9	88	85	E2	85 <b9></b9>
@F58:E3	85	E4	85	E.5	20	9D	2CKEA>
@F60:20	9C	2E	A5	F1	10	05	18<2F>
ØF68:49	FF	69	01	85	D4	A9	00 <f8></f8>
@F 70:85	DS	20	DC	2B	A5	F1	2943F2
0F78:80	Ø5	04	85	D4	20	90	20<71>
3			_ 10-			_	

0F80:A6 F0 F0 01 D3 BD(D3) 0FB8:89 DE 95 FØ CA 3 (2) ES CECERS 4C 0F90:01 D3 86 28 60 18 38(07) 8F 9B: 68 84 **D4** 10 A95673 82 7F 911 A5(20) ØF AØ : 87 2060 49 ED 85 DEAR: 04 29 85 D4 06 FR BDKFBX 31 85 EB BD 37 31 8SKAE) QFB8:E1 BD 28 31 BD 39(28) 85 E2 RECRUITS: 85 E3 RD 30 .31 85 F4.44> €5 OF CB: BD 31 1010 3B 85 28(52) 20 @FDØ:80 C5 A5 D4 7F 38 E9(A6) 29 24 C9 R4 RED8: 48 300 1 (2) RR AAK 2RS OFER: BS 85 F1 D5 63 29 I PI 024R93 18 65 F1 Ø3 65<3B> 0FE8: A7 02 REFREEN 85 EN 84 F.1 09 OLO) 954.985 0FF8:05 10 FB CA 20 BH 20 46<65> 90 9C 1200:FB 89 20 2E 28 BB(FØ) 1998: 2F 26 97 2C 20 0.4 ZE. 28(09) 1010:9B 76 BØ 83 A9 86 A2 TRUSES 1818: A0 31 20 31 2E 23 2F < D2 > 20 18<FC> 1070: 78 9E 26 46 FB 98 09 1028: A5 D4 F0 R4 49 R0 85 D4(F1) 1030:60 BD 83 55 39 14 99 3EKCB> 1038:01 44 27 52 60 BE B1<B1> 92 62<1F>67 40<BE> 1040:75 433 55 SF 07 94 1048:39 59 BE 64 64 28 07 96 32 1050:01 57 40 00<72> 81 74 53 29<15> F8 85 F1 AS(EZ) 1058:00 00 00 T.F. 22 85 1060:25 49 1068: D4 29 ZF. C9 40 20 IB A5<33> Fi ASKRE > 29 80 85 F0 E6 25 D4 85 D4 A2 1070:D4 10:78:7F D4 A2 FA ARKIDS 1080: DF Ø1 D3 EΕ 20 CD 2F CE(39) 9B 26<54> 1088:01 D3 20 D4 2E 20 1070:80 47 A9 BB A2 AE AM DEKEEY 1898: FF 81 D3 28 31 26 CF RICTRO 20 23 2F 18A8: D3 BØ 36 20 9EKBE> 10AB: 24 B0 2E A5 FI FØ 19 A2KF3X 1080:05 FF 01 03 BD FR DE 95<14> 1088:E8 CA 18 FR CE 01 D3 20<35> 1000:90 2C A5 FR 05 **D4** 85 04(22) 19CB: AS FB FD DD A2 85 BD 42(AD) 95 ER CA 10 FR 1000:31 28 08(A7) 1008:28 60 38 60 18 40 PRRC355 10E0:85 F1 10E0:09 3F A5 D4 30 F4 FB F4<36> FØ 85 18 69 21 B5<F4> 10F0:F1 A9 96 85 EF 49 85<29> 3F 10F8: D4 29 D4 2E 20 90 ZE. 20<4F> 1100:BB 2F E6 D5 29 97 2C 20×69> 26 26 A5 D4 BD(1D) 1108:23 28 9E 1118:EC 05 A5 05 8D ED 05 A5(3A) 1118:06 8D EE 85 A5 D7 BD EFKERS 1120:05 A5 D8 8D F8 85 A5 D9<A7> 1128:80 F1 85 20 90 2E 20 FBK.SD3 1130:2E 20KEB> 20 (7)(9) 28 20 2F 3E 1138:97 32 A5 D4 FR(A5) 20 28 4F 1140:0C 20 1E 2F 20 9D 2C C6(84) 1148:EF 38 0.3 10 28 FD 2E < F 1 > 1150:A5 F1 F0 0F 4A 1B 65 D4(51) 1158:69 1F 85 D4 46 F1 23<9E> 90 1160:20 98 32 18 68 40 9C 2B(C7) 1168:A5 D4 29 7F 1170:20 9C 2E F9 C9 86 90 F5<06> 62 88 AR 04(1F) 1179:18 A5 D9 65 ES 85 D9 A5<C5> 1180:08 65 E4 85 D8 A5 D7 65(22) 1188:E3 85 D7 A5 D6 65 E2 R5<2E > A5 D5 65 £1 85 1198:D6 D5 80<94> 1198:69 08 AA 88 D0 DB DB 8A<50> 11A0:F0 1A A5 D8 85 D9 AS DIKESS 11A8:85 D8 A5 D6 85 D7 A5 D5(B4) 1180:85 D6 86 05 A5 D5 C9 IØKEA> 11B8:98 31 E6 D4 45 D5 C9 10<0A> 1100:90 27 48 66 D6 66 D7 6646C2 66 D7<4E> 1108:08 66 D9 40 66 D6 1198:66 DB 4A 66 09 66 D6 66(CB) 11D8:D7 66 DB 66 D9 40 66 D6<E2> 11FR: 66 07 66 DB 66 D9 85 D5(A5) 11E8:60 C6 D4 **A9** 22 D9 86 26(AØ) 11FØ: DB 26 D7 26 D6 26 DS 2A(1D) 09 26 DB 26 DS 2A 06 D9 11F8:06 D7 26 D6<44> 1200:26 26 08 26(47) 1208:07 26 D6 26 25 24 D9(92) 86 1218:26 De 26 D7 26 **D6** 26 D5<41> 1218:2A 60 C9 FF 90 02 CØ 50×CE> 1772:8A 69 888 RS D4 68 98 29,34> 1228:F@ 40 40 40 65 D4(BA) 85 D4 1230:85 D4 98 29 ØF A6 EØ\BD> Dá 1238:50 65 04 85 04 68 AZ 08<89> 1740:04 D5 A5 D4 86 05 38 E9(2A) 90 09 1248:40 CF FB C9 82 FØKEE) 42 1250:45 88 **A5** 06 FØ 4A BE 1238:85 D4 44 4A 65 D4 85 D4<EB> 1268: A5 D6 29 RF A4 D7 F8 59×7F> AA(33) 1268:65 D4 85 D4 98 29 OF

1270:BD CD D4 85 D4 BD(AA) 29(90) 1278: D7 साह 65 D5 85 D5 98 1280:FØ FØ 12 40 AA ASKEB 46 40 7D 65 D5<CD> 07<43> 1290:70 BA 85 D5 60 C0 22 45 07 1298-BB FR 79 FR AA 85(77) 12AB: D4 40 65 D4 A5<95> 40 D4 85 29 2UF A6 50 65(D2) 12A8:D7 DB EØ 1280: D4 85 Da 89 AD 375 65 DAKRON B9 1258:85 D4 B4 2.2 65 05 85<C0> 12C0: D5 A4 06 DØ A7 68 00 10(FE) 60 12CB: 70 30 40 50 20 27 4EK6D> 1200: 75 90 0.5 FA FB W3 DØ 97<B9> 17(F8) 12D8: P8 98 AD 18F 88 13 712 12E Ø: 58 48 1F 28 64(12) IB 23 20 12FB: CB 2C 90 F4 58 BC 20 84(22) 02<36> 12F0:00 22 2.5 07 88 01 01 12F8:03 4C F4<20> 93 28 €5 20 18 1300:20 05 E5 85 A2 84 A3 85(D5) EF 90(87) 1308: A3 F0 98 85 **9A** 05 1310:30 E.8 **A5** 99 85 ARL **A5** 9AKRC) 85<77> 1318:85 AL 20 62 C9 A5 8A 1320:E2 A5 88 85 E3 A9 80 85(AR) 1328:D5 ØA 85 D4 20 ØE 35 30KDC> 88 85 (EA) 1330: C9 AS 89 85 8B AS 1338: BA A0 01 B1 8A 340 44 D8<93> 1340:81 8A 25 9F CB B1 84 85<66> 1348: A7 CB 84 A8 B1 BA C9 MAKEAN 1.350 ER RA C9 BB FB BA C9 BC(F1) C9 1358:FØ 02 0D FØ 69 C9 1E(87) 23 FØ 1360:F0 6B C9 5A C9 B7<66> 1368:FØ 4D C9 84 F8 AR 09 5AC45> 1370:F@ 67 A4 A7 CD<EE> €4 9F 90 8A 1378: 18 A5 65 9F 90 88 E6(E8) 1380:00 00 84 A5 E3 85 89 A5<02> 1.388+F.2 95 SA AR RI RI 94 30K2E> 1390:20 45 9C 91 8A 68 A5 98 (DF) 1398:91 BA 1B 65 A2 65 98 A5<F2> 13A8:90 65 A3 85 90 A0 02 B1(B1) 84 90 DC E6 88 80<72> 1 380:08 20 C5 C9 4C 20<2F> 1788: 66 35 EØ 18 DØ B4 C6 AR< ND> CB C4 A7 1.3CA: A4 AB 88 28<65> AR 1308: D5 34 4C 59 20 66 13DØ:E0 17 FØ Ø7 EØ 18 DØ 9A<84> 13DR: 20 EA A8 A5 AB C5 A7 BRKAA> 13E0:91 48 20 D7 34 68 85 A8(42) 13E8: 20 68 35 4C C2 34 ARKEA? E6 1.5ER: A4 AB 84 DC 81 8A FØ 2E<42> 13F8:C9 9F BØ 4F E4 20(21) 24 26 1400:25 33 A5 D5 30 20 BØ 1E (CB) 1428: 20 GE. 35 DR RD 04 85 D4<57) 1418:84 D5 20 DC 2B D4 28(82) 96 64 AB A2 85 1418:66 D4 85 D4<D6> 1420:88 91 8A CA 10 F8 ADI 05/14) A5 05 E5 1428: D4 C5 99 BØ<ØE> **9A** 1430:05 A5 D4 A4 D5 60 A5 E2430) E3 85 DB 9C 85 E1 1438:85 DA A5 05 98<533 1448:85 FR A5 @1<@1> A/A 1448: B1 DA 50 2D CS DS DØ 05<5E> 1450:88 B1 DA C5 D4 88 15 AØK SB > 1458:82 Bt DA 65 BA B5 DA 90(04) 1460:02 E6 DB BB B1 DA 30 11(57) 1468: 18 A5 EØ 65 42 85 EØ. A5<A8> 1470:E1 65 A3 4C 2B 35 18 FØKE9> Ei 1478:81 38 A5 EØ A4 60 E6<117 20 90 1480: AB 4F E4 ΕB BD< 65> 1488:CC 23 DB F5 60 A9 30 2C< 7A> 8D 16 73 A9 1490:A9 BR FA ARK 7B2 1498:35 BD 6.0 82 80 E.1. 02 A9< 6F > 14A0:04 A0 01 20 C8 C2 EE Ø1<37> 14A8:D3 20 07 36 C9 EE DØ 90(B8) 14B0:C8 D0 5D A9 F6 A0 35 BD< 2C> 1488:E2 02 80 E3 02 20 07 36<06> FØ 14C8: C9 FF C8 FF DØ 84 EB<@C> 14CB: 9D 44 23 98 90 45 03 20<20> 36 48(50) 3B FD 9D 1408:87 44 83 1408:03 98 FD 45 9D 49 93<BB> **M3** 48 03 DO 03 FE 14E0:FE 49 03<10> 30 24 14E8: 20 56 É4 99 20 Ø11<ØD> 03 14FØ:36 AD 53 B3 C9 DØ BB(BD) 20<5F> 14F8: A2 10 A9 ØC 9D 42 03 1500:56 E4 15 10 03 20 ( B4 > 1508:04 36 49 6D 01 D3 60<04> FE 1518: A9 81 AB 28 24 23 78 4CKEE> 02 02456> 1518: E4 DC 6C 6C EØ. EZ 1520: A9 07 AZ. 10 9D 47 0.3 A9(52) 1529:00 90 48 03 9D 49 Ø3 20(06) 1530:56 E4 3D DF 48 A9 MI 90<97> 20 56 A8(55) 1538:48 03 39 D4 E4 40 1548:68 6B 89 79 OF. 24(14)

Listing =Turbo-Basic XL« (Fortsetzung)





1548:E4	AC	25	E4	ĒΒ	DØ	Ø1	C8<7A>
1550:8E	F8	24	80	FP	24	A9	ØØ<56>
1558:80 1560:03	ØE A9	D4 AB	78 80	A9 FA	FE	BD A9	01<28> 24<0F>
1560:00	FIB	FÉ	A9	DØ	BD	FE	FF<32>
1570:A9	24	BD	FF	FF	49	CC	20<4C>
1578: 0B	60	49	EB	20	DB	610	A94 33>
1580:40	8D	ØE.	D4	28	A9	FF	8D<1D>
1588:01	D3	20	01	61	A9	22	85<2E>
1598:80	A5 SF	10C	A4 A9	80 0D	BD AB	9E	20 <ff> 85&lt;52&gt;</ff>
15AØ: ØC	84	10D	A9	FE	80	01	D3<25>
15A8: 85	CA	A2	21	BE.	FB	03	B6 <c4></c4>
1588:89	CA	ΒĖ	44	22	20	24	E6<42>
1588: A9	90	95	A0	85	Al	20	SF (DF)
1500:09	AØ	1F	A2	BA	20	86	25<74>
1508: A0	IE FB	89 20	A7 C5	69	91 A9	97 FF	88<06>
1508: A9	AE	48	20	Cò	60	CE	Ø1<95>
15E0: D3	A9	88	AB	99	02	88	6844C>
1568:10	FA	40	JA	F5	90	20	19<66>
15FØ:07	ØD	67	14	19	25	ØF.	2D<93>
15FB:44 1690:4E	3A 2E	41	55 41	54	4F 16	52	55<15> 88<18>
1608:06	26	16	36	A9	FF	BD	Ø1<3A>
1910:02	A9	CB	85	6A	4A	85	ØF <a2></a2>
1618: AD	81	E4	48	AD	96	E4	48<51>
1620:60	85	03	AR	88	84	D4	A2<8B>
1628:04	A9	FF Z1	80	21 D2	P3	DE.	D4 <c3></c3>
1638:03	89	90	21	91	D4	C8	00(82>
1640:F8	Eδ	05	CA	DØ	E3	60	AZKAC>
1649:19	A9	De	9D	44	63	A9	@@ <ba></ba>
1659:90	45	Ø2	90	49	03	A9	04 <eb></eb>
1658: 90 1660: 20	48 56	03 E4	20	59	PD A9	42 70	85<86>
1669: DA	9D	44	63	89	61	85	D8<56>
1670:9D	45	52	A5	DB	E5	D6	85KEA>
1678: DC	AS	D9	E5	07	85	DD	E9<02>
1680: DC	0.0	02	E6	DD	AS	DC	9D <ff></ff>
1680:48 1690:56	23 E4	A5	DD 2B	9D CE	49	D3	88<20>
1698:00	A6	DD	FB	ØE	Bl	DA	91<02>
16A0:06	CB	90	F9	E6	D7	E6	DB<64>
16AB;CA	DØ	F2	96	DC	FØ	<b>QB</b>	B1 <a5></a5>
1680: DA	91	D6	CS	CA	DØ	FB	A9<8B>
1608:FF	8D	@1 E2	D2	E3	89	99	60<00>
16C8: 80	CO	F7	Ø2	20	02 1E	F4	20 <f1></f1>
16D0: CB	CØ	20	FB	E4	85	AB	84 <fe></fe>
1608:A1	20	FB	E4	20	SF	69	AØ<26>
16E0:01	91	BA	C5	05	DØ	95	GB/EE>
16E0:81	8A 82	19	D4 B1	92 BA	02 A8	65	16<5F> 8A<40>
16F9:85	BA	90	02	26	88	42	BAKAZ>
1700:20	20	26	4C	1.3	CB	28	26(89)
1708; C1	4C	B5	Fl	58	C4	A7	BØ <ab></ab>
1710:03	20	BE	CZ	A9	90	85	89(84)
1718:20	74 20	<b>C2</b>	A5 E4	11 A5	F8	34 C5	A5463> 884F4>
1728:A5	9E	E5	89	BØ	27	A5	89(31)
1730:20	<b>6</b> B	CD	E6	89	DØ	E1	20 56
1738:B1	F3	20	02	F2	A5	D2	C9<2E>
1748: 40	90 92	1E 34	C9	3D	80 20	38 96	C9 <c6></c6>
1750:4C	59	F9	C6	11	A5	85	FØ<6A>
1758:07	20	FB	C4	A9	00	85	B5<62>
1760:60	AB	4F	40	80	ØA	AD	50<77>
1768:4A	19:0	95	49	3F	40	96	C2(EB)
1778:98 1778:AA	20 CB	48	F3	4C	<b>32</b>	B1 F9	D4<8F> A5<7F>
1780: D9	48	A5	DB	48	A5	D7	A6<71>
1788: D6	20	52	F9	49	20	20	96<4F>
1790:CZ	68	100	68	4C	52	F9	BAKBB>
1798:48	A5	90	48	A5	91	48	A5<2D>
17A0: P2	48 03	F7	A@	48 2F	A5 A4	A1	48(A6) 00(8E)
1780:08	A9		85	AØ	A9	CI	85<08>
1798:A1	AB	30	ØE	AØ	22	B1	A@<ØE>
17CH: C8	91	92	B1	AØ	CB	91	90(03)
1708:90	DE	AØ	20	B1	AØ AØ	AØ	13<8A>
1700:91	98	A0	Ø1	81	A0 B5	AØ A1	14(EA) 68(A4)
1750:85	AB	6B	85	82	68	85	91(03)
17E8:85	ØF	86	85	92	85	DE	68 <ec></ec>
17FØ: AA	60	48	A5	90	48	A5	91<24>
17F8:40	A5	B2	48	A5 48	AØ	48	A5(47)
1800:AI	48 9E	A5 48	BA A5	BF	A5 48	20	48 <fb></fb>
1810:C1	20	C5	C9	20	03	F7	86 <cb></cb>
1818:24	48	A4	A1	CB	FØ	05	20<23>
1820:62	C9	90	06	A9	99	85	BA(9E)
1828:85	BB	AØ	01	86	10	Ø2	AØ <a9></a9>
1830:13	A5	8A	71	90	C8	A5	BB(CD)

1838:91 98 4C 48 C1 68 85 8F<84> 1840:68 85 BE 68 85 BB 68 85(3D) FF 1848:8A 4C 10 C1 FF A5 86(F3) 1850:85 AQ A5 87 RS. A80145 A1 A5 1858:C5 88 A5 A1 E5 87 19(BD) 1860:A0 00 81 A0 29 CB C9 DOMAGO 1868: DØ Ø2 91 AØ 18 AS AØ 69( 3A) 1870:08 85 A0 90 E1 F6 A1 86<62> 05 ØF. 1880:98 06 98 06 98 26 25<DC> 9P 1888:98 AA 9A DD 15 A6 99 EØ< 30 > 1890:05 BO OF C4 02 AZKBZ> 90 60 1898:07 A9 00 90 86 D2 CA 10<6F) 18A0: FA 68 4C 28 E5 85 92 CB<2A> 18A8: C4 A7 B0 EB 20 23 E'5 AR< 20> 10 D.3 GO 1880-00 24 62 AD 7B<36> 04 48 1888: C9 BR EA ØA. 80 08(31) 05(42) 18C0:02 A9 03 BD @F D2 20 18C8:E5 68 AA A5 99 9D 86 D2KEED 1800:24 A2 10 07 E8 E8 A5 9A(F1) 18D8:9D 20 D2 A5 98 ØA RIA ØAKB6> 18E0: BA 05 D4 9D 01 **D**2 68 20< DB > 18E8: 13 E5 65 55 B4 56 20 23(91) 18F0: E5 85 54 60 20 13 F.5. B5<10> 18FB: CB 6/0 20 13 E5 80 FD @2<B2> 1980:60 A9 12 ZC A9 48 20(03) 1 3 1908:18 C2 AS CB BD FIB 02 A2< 87> 1910:60 A9 0C 9D 4A 03 A9 00<91> A2(CB) 1918:90 4B 03 6B 4C FE C5 1928:06 86 CI 20 FB C4 20 F3 1928:ES A2 0D A0 23 86 B4(AA) 1930: E4 A2 06 29 FR 49 10 ARKAR> 84 40 99 C4<73> 1938: A5 D4 C3 20 1940:20 18 C2 **A4** CB A2 60 4C<1C> C2<8E 1948:9C C2 A6 B4 D0 03 A5 1950: 20 96 C2 A6 B4 A9 **個**5 20(49) 1958: AD C2 02 C5 4C 189 C4<82> 20 1960:A9 99 AL 85 AB 20 AF DZKFB) 1968: BD 44 83 85 24 BD 48 03<522 BF(E7) 1970:05 28 24 98 4C 20 DE 1978:04 85 CØ 86 G1 4C C4(9E) 85 B4 1980: A9 04 20 C6 40(45) C2 1988: 7E EA A9 0B 20 C6 E2 85(29) 204715 48 1990:85 68 AB 0.7 84 C1 48 03 B4<06> 1998: AP C4 26 FD C4 19A0:00 68 A8 88 20 DC C3 49:16> 1988:07 50 A9 FF 20 A9 292) 48(CB) 1980: A9 04 20 C6 C2 68 48 A9(85) 85 CA 1998:07 85 CØ 20 AB C4<C8> 1900:A0 00 20 04 CS 28 89 C4<665 38<0A> 05 0D 81 05 19CB: AD 80 DØ 1908: AZ 80 18 88 70 00 05 BE 05 1908: A8 A5 B1 7D Ø1 CD E6<28> 82498D 19EU: N2 90 DA DO 05 CC 65 AC 95 94(94) 21 19E8:90 03 **D4** FB 20(74) 19F0:00 CA CA EØ 82 BQ DB 19F8: 6E 03 20 5B F5 09 0501 85(98) 1A00: CA 68 F0 01 60 40 6B E6<30> FB A9(89) 1A08: A9 00 85 CA 4C D1 C2 A9 1A10:08 20 C6 019 85 00(61) 1A18:A2 80 38 85 00 E5 689 9DKAF2 PD(IF) 1A20:00 05 E8 B5 00 €5 81 1A28:00 05 EB €0 DE Eθ 70.52 1A30: AR C4 A0 0E 20 04 CS 1AT8: 89 C4 20 AB C4 A5 87 ASCE95 1848:F3 A5 B3 F4 AC. 80 85 25.28 1A48:88 98 AC 9C 05 C5<2E> 26 86 1A50: 20 B9 C4 4C FB C4 A9 RE(48) 1A58:20 9C C3 4C 48 C3 **A7** @4<53> EA C2<45> C3 A9 00 1A60:20 9C 40 AZ B6(AC) 1A68: 48 A2 10 86 E3 23 1A70:F4 A2 07 68 A8 A9 88 20 (6F) 60(39) 1A78: B4 C3 C4 A9 20 89 07 1AB0:48 A9 23 20 AD 69 90<300 1A98: 48 03 98 9D 4A 83 20 0D(AB) 1890: C5 4C 88 28 20 13 E.5 20(75) C4 1498: A9 03 85 CB as 28<0F3 28 1AA0:13 E5 48 20 13 E5 48 68<67> 1888:48 98 4R A4 AR CB C4 A7 ( 7E > 1AB0:80 1A E4 C5<52) 20 00 20 73 1ABB: 20 AB C4 68 9D 4B 03 68(22) 1ACM: 9D 4A 93 20 02 C4 A9 **C**5 201 937375 1AC8: C5 4C B9 27 A2 23(日取) 71 1AD0: A0 05 20 DA 4C E9 C3<39; C5<84 1AD8: 20 A6 C4 A9 ØD 20 1A 1AE0:98 4C 20 C5 A9 26 20 51(32) 40 03 BC 4D 03 20<21> IAER: C5 BD 1AF 0: 22 €5 20 AB C4 BD 4E Ø3<65. 1AF8:4C 20 C5 20 A6 C4 20 13<84> 4C<45> 9D 1800:E5 20 AB C4 AS D4 D5i 03 13(25) 1909:03 A5 9D 40 20 1810:E5 20 AB C4 A5 D4 4E<37> 9D 1818:03 A9 25 85 CØ 4C S& C5KCB>

1878:20 98 C2 A4 A8 C8 C4 A7<55> 1830:90 F1 A0 B1 8A C9 1.0 FØ(21) 20 C5 1838:0E ØD. C6 28 20 A4<2A> 1840:A8 C8 C4 A7 90 F3 68 20(96) 1848: A6 C4 20 AB **C4** 20 184 C6(65) 1R50: AA 98 20 RF CA RA 70 20<5C> 1858: CS A4 A8 C6 C4 A7 90 EA(8F) 20 C2 A2 ES(AD) 1B68:60 18 60 DØ 1888:81 BA **C9** 10 FØ 04 A9 00<970 78 DD 85 1870:FØ Ø3 2Ø C1 ASKEE> 1978: C1 ØA BA BA BA AA 100 E8K6A> 14 4C FB 20 C4<BB> 1890:A9 DC AB 1B88: BD 43 63 10 D3 AØ 00 8C(FØ) 11(34) C9 80 D0 1890:FE 02 09 84 1898: A6 CA FB C4 4C 5D E6 DACGE > 1840:C1 C9 BB FA DE R.5 89 E9<375 1BA8:07 D0 193 20 F8 C4 20 D1<9C> 1BBØ:F5 4C DE F8 C@ 07 DØ ED(66) F8<01> 1BBB: A2 50 E4 C2 D0 E7 20 1BC0: C4 4C 6E E6 2B AB C4 FØC3F5 1BCB: 97 FF(1D) A9 8C 40 18 C5 AB 18D0: A9 00 9D 49 03 98 9D 4BKEA> 45 1808:03 AS F4 A4 F3 9D 03(4B) 18E8:98 9D #4 #3 A5 CØ 9D 42(EB) 4C 98(0D> 1 PEB: 03 98 24 AB 48 0900 18F8:48 28 00 E4 68 85 D5 68<F70 1868:85 D4 28 DC. 28 4.0 CØ E54.900 9D 1000:A9 00 A2 CA<BD> 07 100 D2 1C08:10 FA AG 07 84 C1 20 FB(AA) 1C10:C4 C6 Ci DØ F9 68 CB E4<40> 1C18: A7 80 E5 A9 0C 85 CR 20(00) CS 1020:A6 C4 20 4C C4<12> 18 B9 1028: AZ 06 86 BD C5 20(C5) 6D 1C30:96 C2 F2 CA F3 86 10 68460> 41 1C30: 9B 59 44 43 52 98 20(29) 1C40: DC E4 A5 D5<47> D4 85 F3 A5 1C48:85 F4 A6 FB **A4** D6 **D7** B2(A4) 1C50: A0 FE R1 E.S. 85 97 R4 98(60) F3 85 92 99 A4(3E) 1058:A9 91 68 1C60: 98 A5 97 91 F3 A9 90 B5(B9) 1068:92 40 86 AB 2B A9 20 C6<68> 1C70:C2 A2 70 20 04 C6 30 074 COL C2 1078:AZ 00 28 9B 10 F2 9B<D9> 1080:48 A2 70 20 FD 68 C9KE6> 1C88:88 F9 03 4C DC F8 68 A9(CE) 1090:20 20 A9 23 A9 20 24 20(07) 21 LC98: A9 95 CØ A9 07 85 C1<4D> 49 1CA0: A9 00 AB 4C DC C3 BB< 20> 29 64 10AB: 20 A9 07 48 96 CA 20(07) 1080:00 E5 A5 28 AB 98 90(50) 90 1CBB: 44 03 A5 90 45 203 A5(4E) 49(41) 1000: D4 9D 48 23 AS 05 9D 1008:03 48 20 C5 18 4C AT C2(A1) 1CD0:A9 07 85 CO AD 88 4C 84<C4> F4 24 CĐ 1C08:C5 20 BØ 80 81<1C> 1CED: 60 4C A9 C2 A9 2B A9(@F) 20 1CE8:07 85 C0 20 90 **C4** 20 6B<7A) Ç4 10F0:E5 A9 28 AB 04 F3<E6> 85 1CF8: A9 00 85 F4 AØ 06 94<CD> 20 1000:C5 20 AB **2B** 28 89 **C4** A5<DD> 1008:C0 C9 07 00 03 20 CØ E5<2F) 60<E0> 1010:A4 AB C8 C4 A7 90 D7 1018: A4 A9 DØ 32 C6 A6 AA(37) 1020:CA AA BD 20 21 20 ED E5(99) 91 41 1028: A@ @2 BD 21 CBKAAS 9D 1030:80 61 21 9D Ce BD B1<5E> CS BD A1 1038:21 91 90 91(56) 21 1040:70 CB BD 9D C1 21 91 CBCFBS 9D 1048: BD E1 21 91 60 A9 BUKFA> 1050:85 B1 60 E.6 BØ A4 A9 6B<CB> 1058: 68 4C 31 E4 00 40 85 B1 < 300 1060:24 B1 10 06 A5 AA 85 AF(C3) 1068: C6 AA A5 B0 AB FØ 07 C6<AØ> 1D70:B0 20 16 E5 30 19 84 98<20> 1D78:85 97 28 16 FS 320 10 85(01) F6 1D80:F5 84 28 6E E5 24 B1(CD) 1088:50 08 A9 99 65 Bl 60 40KFD> 1090:28 E5 46 D2 90 F9 A5 F5<0E> 1098:C5 D6 A5 F6 E5 07 90 EF<BF> 97 1DAØ: A5 C5 DB A5 D9<17> 98 E5 10AB: BØ ES **A5** D9 D0 10 A4 D9<F5> 1DBM-BB DM MB Δ5 FS 85 F7 ASCAR) 1008:FA BS FB 40 14 C7 AB 20<8F> 1000:84 F7 84 F8 AØ 10 46 D9(13) 1DC8:66 D8 90 ØD. 18 65(B4) 1000:F5 B5 F7 A5 FR A5 FA B5<£1> 1008:F8 06 F5 26 FA 88 DØ E6<D9> 1 DE 0: 18 AS 97 A5<05> 1DE8:98 45 FB 06 F5 2A 85 F6<36> 1DF0: AB A5 F5 26 F6 65 F5(A9) 2A 1DF8: AA 98 65 88 65 D4<43> F6 8A IEBB: AA 98 65 D5 AB 8A 65 BC(D3) 1608:85 FS 98 65 BD 85 F6 24(87) 1E10:B1 10 2E A6 AF 86 AA C6<77>

1B20:20 9C C4 20 13 E5 A6 C1(FC)

			_	_		_	
1E18: AA	AB	88	84	21	BD	40	21<79>
1020:91	F5	CB	BD	60	21	91	F5<5E>
1E28: C8	BD	86	21	91	F5	C8	BD<68>
1E 30: A0							
	21	91	F5	C8	BD	CO	51< 4B>
1E 28: 91	F5	<b>C8</b>	BD	EØ.	21	71	F5<6A>
1E40:60	E6	AA	A6	AA	AB	65	B1 <dc></dc>
1E40:F5	90	EØ	21	88	B1	F5	90<3E>
1E 5Ø: CØ	21	88	81	F5	PD.	AØ	21(AF)
1658:88	B1	F5	7D				
				88	21	88	B1 <be></be>
1E60:F5	90	60	21	88	B1	F5	9D<06>
1E68:40	21	SE	20	21	60	A5	80<5C>
1E 70:F0	27	20	Sic	C8	84	78	85<5F >
1E 78:97	26	ØF.	C8	38	E9	01	85(4E)
1680:F5	78						
		E9	98	85	FA	20	PE <ec></ec>
1E 68: C5	95	81	10	ØP	25	80	85 <e6></e6>
1E90:B1	A4	D9	A5	08	4C	DØ	C7 <ca></ca>
1E 98: A5	D6	A4	D7	A6	BØ	FØ	18<6A>
1EA0: C6	PØ	C4	98	90	3D	DØ	Ø4 <c2></c2>
					78		
1EA8: C5	97	90	37	44		A5	97<98>
1EBØ: 38	E5	F5	85	D6	AA	98	E5< 20 >
1E.B8:F6	85	D7	90	26	AB	DØ	Ø3<57>
1EC0:8A	FØ	20	20	DF	E4	16	A5<3F>
1EC8: D4	65	F5	85	D4	A5	D5	65 <c2></c2>
1ED0:F6	85	D5	24	<b>B</b> 1	10	01	6Ø<6D>
1ED8:40	3A	E5	20	16	E5	DØ	F7<02>
1EE0: AA	DØ	F4	A9	05	4C	DC	F8<87>
1EE0:20	DC	E4	A5	D4	85	99	A5<7E>
1EFØ: D5	85	9A	A5	D6	85	A2	A4<23>
1EF8: D7	84	A3	Α4	A9	FØ	8F	A9<@B>
1F00:80							
	85	B1	20	31	E4	A5	D7 <e9></e9>
1F08:A4	D6	26	BI	80	27	20	DC<23>
1F10:E4	A5	D9	64	DB	CS	A3	98×86>
1F1B:06	DØ	68	C4	A2	BØ	04	85 <e9></e9>
1F 20: A3	84	AZ	18	A5	D4	85	9B<60>
1F28: 65	A2	AB	AIII		85	90	65<8D>
				05			
1F 30: A3	AA	3B	99	£5	SC.	85	F9<14>
1F3B:BA	E5	ØD	85	FA	20	50	26<8F>
1F 40: A5	<b>D</b> 2	20	83	E4	38	A5	F9 <f4></f4>
1F48:E5	04	Αĝ	A5	FA	E.5	D:5	AA<74>
1F50:A9	22	25	B1	FD	8F	A9	@@ <a@></a@>
1F5B: 85	B1	E4	D7	90	06	00	<b>05</b> 4B62
1F60:C4	D6	BØ	1.6	60	84	06	86< 7E >
	4C		E5	A4	A8		
1F68:D7		CØ				C4	A7<79>
1F 70:90	Ø1	60	20	00	E4	46	D2<13>
1F 78: BØ	67	28	26	D2	30	65	A5<80>
1F80:F5	69	91	85	F5	85	D6	A5<'D6>
1.F88:F6	69	88	20	54	85	D7	85<92>
	A5						A5<@F>
1F90:F6		97	69	21	85	DB	
1F 98: 98	69	ଥଥ	85	D9	-349	42	AØKEI>
1FA0:00	84	F7	84	F8	AB	10	ASKIA>
IFA8:F5	4A	90	ØF	18	45	F7	65<70>
1FBØ: DB	85	F7	A5	FB	65	D9	85 <f1></f1>
1F98:F8	30	26	66	FB	66	F7	86<87>
1FC0:F6	66	F5	88	DG	E1	6	F5<70>
1FCB: 26	F6	26	15	A6	F6	A5	F5<31>
1FDØ: ØA	25	F6	30	ac	65	F5	85 <bf></bf>
1FD8:F5	AB	BA	65	F6	85	F6	10 <f3></f3>
1FE0:17	4C	2B	E5	A9	88	85	D6<7C>
1FE8:85	D7	A4	F5	84	DB	A5	F6<25>
1FF0:85	09	90	94	CB	20	FØ	E943@>
1FF8: A2	BE	20	88	25	38	A5	97<09>
2000:E5	80	85	D4	A5	98	€.5	8D<41>
2008:85	Ω5	20	CØ	<b>E</b> 5	49	22	AB<61>
2010:A6	F6	FØ	ØA	91	97	CB	D8<9F>
2018:FB	E6	98	CA	DB	F6	A6	F5<85>
2020:F8	06	91	97	CB	CA	DØ	FAK543
2028:40	AØ	CB	28	C5	C9	A5	8A<47>
2030:85	BE	A5	BB	85	BF	A5	A1<9A>
2038: AA	ØA.	AB	B1	80	85	SB	CB <cb></cb>
2040:B1	80	85	84	AØ	21	84	D1<8F>
2048:8A	00	13	88	B1	BA	C2	A@<82>
2050: RO	13	AG.	82	B1	BA	65	8ACCB>
2058:90	EB	E6	88	80	E4	98	03<77>
2060:18	90	EF	38	60	DB	FC	18<34>
7068:60	A9	C1	20	A9	C2	AA	C8<67>
2070:81	99	26	03	CB	BI	99	49<1E>
2078:80	20	ED	ES	8A	Di	9D	F@<87>
2080:33	91	9D	A5	99	AB	02	91(88)
7088:9D	CB	A5	9A	91	9D	4C	E8 <ce></ce>
2090:09	A5	99	48	A5	94	48	A9(B9>
2078:00	AB	91	88	CB	C8	DØ	FACF5>
20A9: A5	89	85	98	A5	88	85	99<7A>
	84		99	C9	4F	FØ	B9479>
20A0:A0		BI					
20B0:C9	SD	FØ	P8	AB	01	B1	99<21>
2089: 0A	80	18	A8	B1	80	DØ	89(F7)
2000: AS	9A	91	80	CB	A5	99	91<18>
2008:80	18	AØ	62	81	99	65	99(86)
20D0:90	D4	E.6	7A	B49	DØ	A5	88<33>
2008:85	99	A5	89	25	94	AB	QQ< 18>
20E 0: B1	80	DØ	<b>8</b> C	A5	94	91	B2KC4>
20E8: C0	A5	99	91	80	4C	2B	CAKE72
2050:85	9A	CB	B1	82	85	99	C8<7B>
							99(EB)
20f B: D0	E6	48	85	9A	98	85	
2100:60	A9	1E	4C	DC	FB	81	8AKAF>

2188:C8 C4 A7 B8 1A C9 1C DC(FB) 2110:1A E6 AB 20 4F E4 A5 D2<ED> F.5 AR 88 B1 D4<AB> DØ 2170:AA CB 81 D4 AB 8A A9(69) 2128:00 AB 60 40 13 E5 20 MAKARD 2156.CA CS 56 30 84(96) 27 85 9.7 2138:BB A9 00 85 86 60 4C 28(00) 2140:E5 A5 B7 B5 A0 A5 BB 85<51> 7148:A1 20 62 C9 85 A5 8A F3(87) 2150:A5 88 85 F4 A5 BE 85 BAKA92 2158: A5 BF B5 88 A6 PMP B1 F3(47) 2160:85 B7 C8 Bt F3 B5 B8 CR(FR) F3 85 2168: B1 F5 84 F2(11) F2 2178:CB B1 F3 85 F6 CB 84 F2<43) 2178: Bt FX C9 M1 FB 23 A4 FAC425 2180:C4 98 E9 F5 B0 03 88 88(88) 2188:84 F2 A0 Ø1 91 38 42<205 2190:38 A5 F2 A5 F3 R5 F3 A9KE9 2198:00 85 BA 65 F4 R5 F4 904 760 71A8:88 A9 96 85 F5 A5 F5 CSCDCS 21AB:86 88 12 E6 F2 A4 F2 B1(45) 21B8:F3 C9 98 FR 05 C9 2C DR(FR) E6 F5 440 2:88:F2 B3 80 85<F4> A9 7100: A6 E6 F2 4C 42 CB A4 F2(38) 2108:B1 F3 C9 20 18 FM 02 E9(CR) 21D0:9B 60 A9 06 4C DC FB A9<DB> 2108:3F 85 C2 B1 8A C9 @F DØ(8B) 21EB: 12 20 86 E4 20 DD CB A4< DB > 21E8: AB E6 A8 B1 BA C9 12 DØ<5F> 2150:02 66 84 20 4F E4 C6 A8(42) 21F8:90 85 28 78 DD 85 B4 20(53) 2200:AB 28 28 7E C2 20 5B CB(5B) 2208:A0 26 84 A6 84 F2 28 4F<SA> AB A5 02 2210:E4 E6 30 23 29<843 2218:C8 28 RØ 15 20 FA CA DØ(9E) 20 CB ES 4C AC CB 2220:18 ASK ØA> 2228:11 FØ Ø1 60 C6 4C A4<C4> 2230:F5 A9 00 BS 84 A9 08 4C(05) 2,238: DC F8 A0 20 A9 A2<(C3>) 11 8D 2240:20 84 A9 84 88 B4 AA 84<6B) 2148:81 20 3A ES C6 F2 A5 F2<34> 2250:85 F5 A7 FF E8 EA F2 A4(1E) 2258:F2 B1 F3 C9 9B F6 28 C9(2A) 2260:2C D0 F1 24 A6 58 ED A4<FB> 2268:F5 A5 A8 48 8A A7 F3 29< 44> 2270:8B E4 68 85 A8 CB<84> 20 1F E6 2278:24 A6 59 11 B6 A6 48(5C) 2280: EB E4 A7 RO DE 28 FA CA\38> BE CA A6 AB E8(84) 2288:90 18 40 2290: E4 AZ 90 28 20 AB ZB A9<D2> 2298: 80 85 84 68 20 FA CA 98<C7> 33 90 22A0: 03 4C CB E6 F2 4C 42<5C> A5 D4 90 22A8: CB A2 44 83(E1) 2280:A5 D5 45 03 A5 D6 90 90<9B> 2288:48 03 AS D7 9D 49 B3 A9<3C> DF<DA> 2208:08 4C FE C5 DB F9 80 2208:20 25 E5 85 A2 4C(F6) 22DM:5D 26 28 85 F5 85 A2 84(A9) 4C DE 20(44) ZZD8: A3 25 20 9B E5 22E@: 9D 20 4C E5 20 9B ESKBE > 4C 26 2258:20 97 20 36 ES 20 9B<4C> 4C 36 E5 20KF9> 22F0:ES 20 9F 98 28 22F8: 99 E5 26 4C 36 E5(82) 49(53) 2300:A6 AA BD 40 21 F0 05 90 21 60 20 DC DB<9A> 2309:88 40 2310:90 62 FB 60 BB 50 70 DC< 953 2318:08 FØ 4B D0 57 28 DC D8<77> 42 20 2328:98 52 BØ DC DB 98<6C> 2328: 3D FØ 3B 88 47 28 DC DR4.355 2330:90 34 40 20 DC DB 80 FR(15) 2338: 39 DØ ZB C6 AA A6 AA C6(56) MAKSC) 2349: AA RD 48 21 3D 41 21 2348:F0 1C D0 28 C6 AA AA AACDR> 21 2350:C6 AA 8D 40 1D 41 21(2F) 2358: 0A F0 WE DO 17 A6 AA C6(56) 2360: AA BD AR 21 FØ ØF A9 700<4E> 236B: AB F0 AØ 83 2ID 2C AB 82(54) 2370:A9 40 DØ 24 A9 48 A0 @i<DB> 2378:E6 AA 86 AA 9D 40 21 98(38) 21 21 2380:90 60 A9 80 9D 80 21<7F> 90 AØ 21 9D 2388: 9D 88 CR< 97> 2390:21 9D ES 21 60 A6 AA C6<51> 2398: AA BD 48 21 ØA FØ C7 A3<16> 23A0:88 6A DØ D2 A4 A9 B9 2348: 28 C9 ZF. 90 05 40 25 D9(8E) 7380: A4 AA AA. A6 AA C6 C6 AAK DOD D9 2388:8D 40 21 40: 21 DØ 20(79) 2300:0A 98 82 E8 88 80 60 21KFF> 23C8: D9 AB 21 21(FB) DØ 2.F BD SB 23D0:D9 21 DØ BD AB 21(CB) 80 16 21 2308: D9 A@ D@ ØE. BD CØ 21(0B) 06 23E0:09 C0 2: D0 23E8:09 E0 2: 60 23E0:09 C0 BD EC 21(A7) 19 40 21 10(DB) 23F0:FA 6A 49 88 28 68 28 DC(A5)

23FB: E4 20 9C 2E 20 DC FA A0(AA) 2400:00 AS D6 D0 06 AS D7 F0(CB) 2408:05 C6 D7 C6 D6 AA 08 A5(A6) AS EJ FØ 2410:E2 DØ BA. 20 C6(6D) 5.6 2418:F3 28 FR D4<1F> 2420:01 E0 D0 06 E6 D4 D0 02(C5) 05 242B:F6 E6 E.Ø DØ 03 E6 E1(09) 2430:D0 CF Er(2) 02 CB 18 28(81) 2478:38 60 20 DC E4 A5 D6 64<795 7440: D7 D4 B4 85 D5 20 DC 28: 6A) 92 2448: A9 85 D2 85 36< 7D2 2450-F5 20 16 E5 AR RR B1 D4<560 79 2458: 4C D9 C6 AA 38 AD E5<19> 2468:82 E5 90 AA AD E.6 92 E5<85) 2468: 91 AB BA 4C 79 D9 20 73<5B> F2 2470:C5 A9 ØØ 85 20 CR 2B(AR) 2478:20 93 90 CB 4C D7 F8<74> 2480:20 DC E4 A0 00 B1 D4 2488:79 D9 20 73 C5 A0 00 4CKFC> [25/2] PACAR'S 2490:F2 84 D4 84 05 EA<870 20 68 2498:90 FB 20 93 C5 4C 70 D9<80> 2440:28 DC FA 4C ZD. D9 Δ9 MAKE 33 24A8:F@ 8A A9 98 D0 86 99 ØC<FC> 2480: DB 82 A9 14 85 EC 28 16(01) 24BR: F5 DR RF C9 88 BB ØB 65KCF> 24CB:EC AA ED 7B 02 AB 88 40<62> 24C8:79 D9 4C 28 E5 20 28437> AB 2400:A0 00 A5 D5 F0 13(47) 03 20 24D8: DA A5 D4 48 4A 40 44 4AKCB> 248 0: 20 1E DA 6B 29 8E 09 30<39> 24E8: C9 3A 90 02 69 26 91 F3<06> 24F0: C8 60 20 16 E5 20 05 DAK 255 24FB: A9 80 DØ 3B 20 AE ES 28<F2> 2500: PE 2B 2508: FB 29 2B A0 FF CB B1 10<CE> 7F 91 食玉 CR AS FJKE5> 25 AD 2518:08 FC 02 AØ C9< 3C> 99 2518:CØ BØ 18 A2 ØE DD DC(C4) 28 80 2520:FB 13 CA 10 F8 C6<81> 2528:4C 6B DA 70: 16 FS A5 DAKCCO 2530:80 C0 05 AØ. 01 CØ A9 A2<69> A9(AØ) 2539:05 86 D5 B5 D4 84 D6 2540:00 85 D7 85 D3 A9 78 BSKE4> 2548: D2 4C AA A9 30 E5 C6 3F(C9) 2550:85 D4 A2 Ø5 AD 29(31) @A D2 2558:F0 C9 A0 B0 F7 85 EØ ADKR2> C9 2560:0A D2 29 ZIF ØΔ BØ F7(FF) 2568: 05 FØ 95 **D4** CA DB 20<94> E5 2570:88 ZD 4C 80 2578:20 Z6 D8 4C 2580:8D 40 Z1 29 09 20 DA<91> 86 FF DA AA AA(74) 7F 21(F7) 90 40 7FKEC> 7588: 50 29 6E E5 A5 **D4** 29 2590:38 E9 40 90 14 AA A9 MAKEAN 95 90 82 AZ 94 2598:E0 05 DS<41> 25AØ: CA 10 FB 28 68 20 3A< 783 60 20 D4 29 25A8: E5 E6 AA 2F 7F DB 28(A7) 2580: 6E E5 A5 D4 38 E9(79) 2588: 40 90 49 AA A9 00 M4 (DA) E@ 2500:80 3D 95 D6 2508:66 E5 A5 D4 EB 98 E7 20(31) 29 7F 38 E9<130 99 2F 2500:48 AA EЮ 24 DI<CD> 88 2508: A9 00 AB 15 D6 94 D6 EBKCF > CERC: FO DA DO FT 20 D4 10 1762C) 25F9: AB FØ 14 CØ 85 A9 A8(92) EØ 25F0:01 84 E1 68 84 E2 94 E3(C9) 25F8:84 E4 84 E5 20 2600:3A E5 06 D4 20 9n 4C<170 20 90 2R 90KEAD 2608:F6 80 E8 20 6E E5 20 80(40) 2610:50 40 38 ES 20 ЬE E5 20(41) E5<99> 2618:84 SR 4C 38 E5 20 6E 4C 2620:20 4B 31 38 £5 20 2628:E5 20 EE 2F 4C 28 E.5 20<C4> 2630:6E E5 20 F2 2F 4C 38 E5<97> 20 44 26.08:20 6E E5 2F 4C 38(74) 2640: E5 20 AE 20 C5 31 ES 4C(29) 2648:38 ES 4C 9E DB 4C AC DB< /C> 2650: A5 E0 10 F6 AC. 28 F5 20KB5> 2658:98 E5 A5 EØ FØ EF A5 D4<A6> 2660:FØ EE 10 26 29 7F 85 D4<AB 2668: A5 EØ 29 7F 38 F9 40 98(BB) 26/0:E3 A2 04 C9 04 F0 0E RDC AF > 2678:11 AA AB B9 E2 D4<03> 00 D0 2680:C8 C0 64 DØ 85 2688:80 01 18 A9 00 6A 48 A5<55> 2690:ED C9 40 D0 4D A5 E2 BSCE95 2698:E3 05 E4 05 E5 D0 43 ASKE6> 26AØ:E1 29 F8 40 85 FC 44 44 26A8:65 FC 85 29 RF (AR) FC A5 E 1 26BB: 65 FC 85 FC 20 85 ZE 45(14) 2688:FC 90 03 20 BB 2F 20 D4<2F> 2600:2E AS FC FØ 50 2E<E10 26C8: 20 98 26 80 0F 4H FC 90<A0>

Listing »Turbo-Basic XL« (Fortsetzung)

Т								
	26D0:F7	20	95	28	20	23	2F	20(25)
	2608: 9E	26	70	€2	4C	2E	ES	4C< 5B>
	2650:28	E5	A5	E5	48	A5	E4	48/81>
	26E8: A5	E3	48	A5	Ę2	48	AS	E1<@D>
	26F0:48	<b>A5</b>	E0	48	20	F2	21	BOKEI>
	26FB: E6	68	65	E.0	6B	85	E 1	68<26>
	2700:85	E2	68	85	E3	AB.	85	E4<14>
	2708:6B	85	E5	28	9E	26	BØ	CCCC2>
	2710:20	59	2F	82	C7	68	10	Ø4<50>
	2718:05	04	85	D4	4C	3A	E5	A9<69>
	2720:06	2C	A9	88	85	FB	60	20<98>
	2728:16	E5	85	E8	84	E1	4C	16<5F>
	2770: 55	20	5F	DC	98	25	E1	A8- 452
	27381A5	EØ	25	D4	4C	79	99	20<40>
	2740: SF	DC	98	85	E1	AB	A5	E8(51)
	2748:05	04	4C	79	09	20	SF	DC<11>
	2750:98	45	E1	AB	A5	EB	45	D4 <dc></dc>
	2758:4C	79	D9	98	98	9D	9B	B2<0B>
	2760: 85	86	B2	A6	3C	70	BC.	27 34
	2768:67		20	13	E5	A5	14	A4(D3>
		A7						
	2770:11	FØ	11	CS	14	FB	FB	AS(A7)
	277B: D4	DIZ	04	CA	05	38	05	C9<88>
	278Ø: D4	4C	45	DC	96	A6	AA	CA <a6></a6>
	2708:E6	AA	E6	AA	18	BD	66	21<84>
	2790:9D	92	21	80	01	21	90	03<7B>
	2798:21	AB	69	20	AA	90	EE	40KB8>
	2700:20	BD	DC	20	E4	DA	20	26(98)
	27A8: D8	4C	1D	DB	A5	C3	4C	FD(EØ)
	2790: D9	A5	BA	84	BB	4C	79	D9<04>
	2788: E&	C9	A5	C9	FØ	FA	65	AF(FE)
	2700: A9	00	85	94	A4	AB	81	8A 85>
							FO	7A<2A>
	27CB: C9	12	FØ	4C	C9	15	FB	6D <d4></d4>
	2700:09	14	FØ	76	C9			
	27DB: C9	1.0	FØ	5A	20	50	E4	C9 <b2></b2>
	27EØ: AB	Ab	AA	BD	90	21	38	83 A4 -
	27EB: 20	34	DA	28	DC	E4	A6	BS <f7></f7>
	27F0:20	AF	C2	A5	D4	PD.	44	<b>8</b> 3<67>
	27F8: AS	05	9D	45	@3	A5	D6	9D <a9></a9>
	2990:48	20	18	65	94	85	94	A5-89>
	2800:07	9D	49	03	05	D6	FØ	B4(99>
	2810:A9	ØB	20	FE	C5	4C	FÇ	DC<86>
	2818: A4	94	CB	04	AF	98	89	18<32>
	2820: A5	C9	65	AF	85	AF	70	
	2828: A4	94	C4	AF	B2	17	20	02 <ee></ee>
			94	4C			20	78<09>
	2850:F2	EA			60	DD		
	28 t8: DD	85	B5	C6	A8	4C	FC	DC <cd></cd>
	7840: 86	A8	4C	13	€5	E6	AB	4C <ca></ca>
	2848:FC	DC	Α4	AB	88	B1	BA	C9<6B>
	2050:15	FO	87	C7	12	FB	03	20<0A>
	2858: 94	C2	A9	66	85	85	60	A9<83>
	2860:13	85	F3	A9	23	85	F4	C6<21>
	2868: DB	A2	07	86	B5	A9	99	A9/89>
	2970:08	20	<b>B4</b>	53	20	89	C4	20:06/
	2878:F2	DC	40	FB	C4	A9	3F	20(91)
	2880: A7	FF	95	DF	64	BØ	88	98 (ØA)
	2888:FØ	82	20	16	E3	85	DA	84<69>
	2890:DB						2E	
		£4	18	A5	D4		DA	85(96)
	2898: DC					65		
	28AØ: D4	A5	05	45	DB	B5	05	BØ(47)
	28A8: 22	38	A5	06	E5	00	85	D6 <f9></f9>
	2880: A5	D7	E 5	DB	85	97	90	13<8E>
	2068:05	06	F@	ØF	38	A5	26	E5KFQ>
	28C0:E2	83	DC	AS	D7	E5	E3	85 <e5></e5>
	2808:00	86	60.2	4C	9E	DB	E 6	DA (CE)
	5800: DØ	02	E6	DB	AB	68	B1	E0<39>
	28D8:51	D4	25	DF	D-G	36	<b>A5</b>	D4 <bc></bc>
	288 0:95	80	A5	D5	85	09	A5	E@ <a5></a5>
	28E8:85	E4	A5	E1	85	E5	A5	E3<70>
	28FØ:85	DE	A6	€2	EB	CA	DØ	Ø4<18>
	28F8:C6	DE	30	25	B1	08	51	E4<68>
	2900:25	DF	00	ØA.	CB	DØ	EE	E6 76>
	2908:09	E6	£5	40	20	DE	E6	D4<32>
	2910:00	92	E6	DS	A5	DC	DØ	04.00
	2918:C6	DD	30	AF	C6	DC	4C	Ø6 <b1></b1>
		A5	DA	Α4	DB	4C	79	D9 <d7></d7>
	2920: DE	06	85	D7	AD	88	16	
	2928:85							E6<67>
	2930:36	£5	26	06	90	28	1B	R5<8E>
	2938:66	65	D7	95	E6	90	62	F6(B2>
	2940:E5	88	DØ	EA	60	4C	28	E5<2F>
	2948:20	05	E.5	00	F8	48	A4	AB<57>
	2950:08	C4	A7	84	05	20	13	E5<37>
	2958: DØ	EB	68	90	02	AP	81	B5<91>
	2960:D5	A2	16	A9	99	95	06	CAK9A>
	2968:18	FB	A5	D5	85	EB	EÐ	20<1E>
	2970:60	DE	A5	04	A2	02	20	68(11)
	2978: DE	AS	05	85	D6	40	88	Ø6<2D>
	2980:DC	26	DP	26	DA	186	D6	90<32>
	2988:11	18	A5	DC	65	E8	85	DC<8D>
	2990:A5	DB	65	E7	95	DB	98	82489>
		DA	88					
	2998:E6			20	E2	20 DF	37 20	DF<10>
	29A0: 20	50	DF	20			DF	6A<4E>
	29A8: DF	20	4E	DF	20	50		201762
	2980:4E	DF	20	-6A	DF	24	E2	30<06>
	2988:21	E6	E1	18	A5	ΕB	65	E9<38>

2900:85 ER AS EA 65 ES 85 EAK 74> E6 E9 38 A2 02 B5<4B> 2900:E2 F5 E9 95 E2 CA 10 F7<94> 29DB: 30 CB AS FR F (2) 2E C6 FREAFY DC E5 A5<33> 29E8:38 A5 EB 85 DC 29E8: DB E5 E7 85 DB 89 02 C6(89) 75 29F0: DA 18 A2 92 **B**5 DAK83 29F8:95 E2 CA 18 F7 338 9F 18(12) 9A(DB) 2AMM: AS 99 AS FM 85 55 A5 89 86<8E> 2A08:69 00 85 56 68 A2 2A18:54 86 55 86 56 68 38 ASKEC: EØ 85 55 **A5** 94 E9<8B) 2A18:99 E5 2A28:00 85 56 60 18 65 9B 454 330 9C FB(59) 2A28:E1 85 34 A5 69 120 93 E1 85<A4> 2A38: 8E 60 38 **A5** E3 F2 2A38:54 A5 9C E9 88 DØ A4(A9) 2A40: CB A2 60 28 DF 24 CB RECEED 2A48: 00 E7 4C A9 C2 84 96 A2<61> AF 95 F2 AB- 07> 2A50:00 B6 85 A6 2458: 20 B1 95 FØ 27 BD 68 2A68:29 7F E8 51 95 239 DBKF50 DØ 2AA8: DR F3 86 F8 155 20 86 CB(DD) 95 16 FB ΔE 2A70:B1 FB(9A> 28 E6 2A78:00 98 65 95 90 D5 Εá 96<90> 2A80: 80 D1 18 68 38 68 49 04<7E> 2A88:4C DC F9 A2 20 86 AF B4<CA> 2A90:96 85 95 A6 AB 81 B1<92> F2 2A78: 95 E9 BD FB 88 85 29 7F < 8D > 2AA8: EB C9 2E F0 0b 51 95 DB(F5) DO EF FB 89<BF> 2AA8:03 CB BA D3 2AP0:05 CB **B**1 95 18 FB E6 96<31> 2A98:38 98 65 95 90 D3 E6 2ACO: BA CE 20 E4 E3 FF 90 00<310 84 49 84(C9> 2ACB: A9 8D A2 20 11 2ADD: 80 84 AA 84 BI 20 4F E4<632 2AD8:90 06 28 30 E5 40. **SE** F4(87) 2AE8:85 AB AA 80 CC 23 83 AC(62) 2AE8: A4 A9 BE A2 20 28 24KC60 BĐ 2AF0:C5 AC ØE AA 90 FR 16 B9<562 2AF8: A2 20 C6 A9 28 48 E4 4CKBE> 2800:22 E4 A5 AB CB 20(05) 99 **A2** 2808:84 A9 40 ØF E4 ØA 8D<02 23 A4 5A FB 2918:40 E4 6C 2F AB E6(E2) 2818: AR R1 RA 3.0 53 C9(F3) FØ 27 CB B1(3C) 2820: OF 500 98 68 2928:8A 85 04 C8 91 85 D5<49> 86 2830: CR R1 BA 85 D6 CB B 3 BA(53) CB(F9) CB BA DB 2038:85 D7 B1 85 2840: B1 8A 85 D9 CB 84 AB A9<46> 2848: 88 85 D2 A8 CB B1 GB. A2(FE) 2850:8A 85 D6 85 D8 C8 98. 18<30> 2858:75 90 85 D7(7E) 85 D4 A9 86 2860:85 09 75 81 85 98 2868:06 A8 A9 83 85 D2 64 ABKE32 2870: 18 AB CR EA AR R1 BA 49(AB) 2878:80 85 D3 20 ED E5 9D<43> B1 2880:85 DZ A0 02 91 D4<7E) 9D 85 2888: CB B1 90 85 05 CB 31 90<0E5 2890:85 D6 CB B1 9D B5 D7 **€8<17**> 2998:B1 910 85 DB C8 B1 90 8549CX 24(48) 28AU: 09 60 28 E.3 62 28A8:02 00 15 05 02 65 02 46(61) 2890: 96 3F 18 A5 D4 65 BC B5(2B) 2988: D4 AB AS 05 65 BD D5 ( 9F ) 85 2BC0: 60 20 13 E5 10 FA A9 07<CF> 2RC8: 4C DC FR 28 13 E5 65 99(40) 84<F6> 28D0:84 98 20 13 E5 B5 9B 90 28D8:9C 20 **E4** 20 6E E5 28(23) 288 0: 25 33 80 88 AS D4 A4 D5< RED 2BE8: 60 20 13 E5 F0 DZ A9 03<42> A9 89 2050:20 2¢ 49 ØB 20 494 E3 EE<8C> 28F8: 0A 4C DC F8 80 F6 BR 2000:E6 AA A4 AA C0 76 BR EF DID 99(FE) 2008: A5 D9 99 60 21 A5 D8 2C10:C0 21 D7 99 AB 21 2018:06 99 80 21 A5 2028:21 A5 D4 99 40 99 A5 05 68KAF > 21 AS. 0.5(96) 2028:99 99 20 A5 02 22 21<05> 21 2030:60 20 00 E4 A4 AA **C6** AA CB 2C38:89 E0 21 85 D9 89 CAL 21(1F) 2C40:85 DB B9 AØ. 21 B5 D7 B9 E43 2048:88 21 85 DA B9 68 21 85<45> 87 2C50: D5 B9 40 21 85 D4 20, 68) 2058:21 85 D3 89 00 21 85 024082 A4 AA B9 21KE62 2C68:68 C6 AA E1 2C68:85 E5 **B9** C1 21 2C79:A1 21 85 E3 89 81 21 85(07) 2C78: E2 89 61 21 85 E1 B9 41 (CA) 85 EØ 40 70 D3 (BB) 2080:21 85 E.5 2088:20 ED E5 A5 D2 91 9D C8(E9) 2098:A5 B3 91 90 CB A5 D4 91<E5> 2C98:90 CB AS 05 91 90 CB A5<4B> 2CA0: D6 91 90 C8 A5 D7 91 9D (80)

2CB0:91 9D 60 BA ZA 2A 20BB: 6A 29 FB 18 65 86 R5 90(E7) 2008: 9B 29 07 45 87 85 9F ARKE93 92 2008: 80 60 A9 00 85 85 CAKBED CCDD: AD E7 92 AC E8 85 2CDB: 84 81 CB 85 82 84 B3 85<7F> CCEM: 84 64 85 18 69 21.1 98 81 (DA) 2CEB: CB 85 B6 B4 B7 85 88 84<39> 69 84 BB 19 03<F05 2CFØ: 89 85 8A 2CF8: 98 CB 65 80 BD 2000: BE B4 BF 85 90 94 91 R5482 > 2009: ØE 84 RF A7 RR AB. 91 84(FA) 91 8A C8<BF> 2D10:91 CB A9 80 8A 2018:A9 03 91 8A 89 ØA 85 C9(99) 2D29:4C 66 **A2 9A P8** A5<48> 2D28:CA F0 03 20 04 E6 20 F7(23) **E5** 20.39: FA 20 25.4 C5 28 D1 A5<330 93 2038:92 5C(B4> FB 83 20 C5 20 CA DØ 2D40:C5 20 CB CØ A5 APKRE> 2D48: A2 FF 94 DB 20 AB 28 2050:50 R5 C2 20 R7 E2 0.5 11<800 04 Ca 2058:00 11 DØ AB **BRCRAY** E6 2060:84 F2 84 95 84 94 84 A6KES 2068:84 B3 B4 80 84 91 **A5** 84(9D) 2D70:85 AD AS 85 85 28 10(90) AE 2079: EB 20 FB E7 20 F7 E8 A5<6F> 10 82 85 A6 28 EB(AC) 2080: D5 SD C9 2088:84 AB BI F3 99 DØ @7<2B> 2098:24 A6 38 80 4C E0 E7 A5<B4> 2098:94 85 A7 10(69) 28 F7 EB 20 2DA0: EB 60 EB A9 50 28 DE(FB) 2DA8:66 EC 38 98 AS ΔF C9 15<3D> 2080: DO 23 86 DA AR EC. A9 RØ(11) A9 65(1A) 2088:20 C3 DE 36 80 8.6 PUED: AF 69 98 02 10 24 3845E> 2008:00 A9 BF 85 95 49 FR 85(10) 2009:96 A9 15 A6 DA 86 F2 28<75> 2008:F7 84 EB(E2) EB 20 28 21 20 9F 2DE#: 98 35 **B**1 C8 91 F3 20E8: DØ 06 88 49 20(FR) F3 A9 A6KZE> 2DF0:09 86 91 40 05 20F6:85 A6 **A4** AΘ F2 03(67) 84 **A2** A9 20<FB> 2E00:86 A7 E8 86 F2 94 2F 08 - F7 FR ΔA B1 F3 EA F247C> 2E18:C9 F3 EB A5(32) DØ F7 98 20 99 2E18:94 94 A7 82 21 **A4** F2<65> 2E20:88 81 FØ 03 4C<7E> F3 09 9B 2E28: D1 E6 AB **82 A5** 94 99 00<050 26.30:21 SE **C9** 00 80 04(56) 20 A9 2E 38: A8 82 B1 BA 94 FOKAB> 2640:24 BØ 13 49 EE. AB CB A2<4B> 25 2£ 48: 8A 20 86 A5 97 85 BA(09) 2E 50: AS 98 85 88 00 ØF 18< F@> A8 88<F6> 2E58: 45 8A 83 8A 90 82 E6 20 21 2E60: A2 BA 28 26 04 94 BR(AS) 91 25.68: B9 99 BA 98 100 F7(9F) 50 2E70:24 46 2A A5 **B**1 BA **BAKEA** 2E78:0A AB A2 89 20 20 38<965 26 21.80: A5 84 85 AD AS A5 85 E5(39) 84 A6<4A> A2 20 27 24 2E88: AE 26 31 2E 90: 18 86 20 2£98:20 07 F2 4C 7E E6 10 FRKEED 4C 2EAD: 28 26 C1 AF FF 20 **5**F<F3> 2FA8: C9 PØ AD 22 81 AB( AE ) FØ BA 2E BØ: 18 65 BA 85 BA 90 02 E6<77> 2668: BB 02 RA 20 28 26 AC. 7F(45) 20 CB 28 98 28 Α9 RECO: EA **200**< 890 80 2EC8:85 F2 AB 38 68 28 25(BA) 2E DØ: 13 RØ 63 94 D5 30 EF A5 \ B6 > 25 DB: D4 B4 A1 85 AP 20 F7 E8KFF> TELD: A5 A1 85 D5 40 E.B AR- B2> 2EE8:00 **B4** A9 81 95 BA R9<7C> 2660: 39 60 85 9D 8D 82 84 89(9A) 2EF8:3A FØ 9E 8D 83 04 A5(17) 85 2F00:94 BD 81 94 A5 F2 8D 2F08:04 DØ 02 E6 A2<19> E6 9D 9E 26 1 D - 000 A3 SD .508 SHC: C9 95 98K 4A> 20 E6 E8 90 EB 4C BC<65> 2F 18: 4F 2F 26: E8 ØA. AΘ **B9** 3A FØ 48 2F 28: 39 FO 46 C8:12 B9 0E AR(nn> D2 2F30:A8 68 20 74 E8 90 4C<9D> ĘΒ 48 99 48 68 A9(25) 2F38:8C **A6** FØ 2F 40: E8 E8 E8 EΒ 1F APKBE > 2F48: A5 F2 9D 80 04 A5 94 9D(FA) 2F50:81 D4 A5 9D 9D 82 184 A5<76> 2F58:9E 9D 83 84 69 9D 68(90) 2F60:85 9E AC: 43 FR 4C 00 ESCRE) 2568: 86 A9 81 68 80 94(E6) DØ 82 2F70:85 9D SD **B**3 24 85 9E CA4853 2F78: CA CA CA 86 A9 80 03 40(29) 2F80:43 E8 E6 9D D8 R2 EA 9EKR95 2F86:A2 88 A1 90 30 F4 59 2004 7B> 2F90:F8 D6 B0 EE A5 F2 C5 9FK50> 9F A6 A9 BD 8048B3 2598:90 02 65

2CA8: CB AS D8 91 9D C8 AS D9(8F)

_		_		_					
	2FA0:04	85	F2	BD	вι	94	85	94< AF /	329
	2FA8:4C	43		60	C9	ØF	DE	41<43>	329
	2F80:E6	9D	DØ	82	63	9E	A2	86<9D>	326
	ZFB8:A1	7D	18	C6	94	A4	94	99<6A>	32/
	2FC0:00	21	E4	94	DØ		A9		326
	2FC8: 4C	DC		A2	FF	9A	A5	94 <ac></ac>	328
	2FDØ: A4	A7	99	86	21	4C	D1	EA <de></de>	320
	2FD8: A2	FF	9A	AØ	94	89	22	21(30)	320
	2FE0:C9	54	DØ	QA.	84	F2	88	A9 <f7></f7>	321
	2FEB: 9B	71	F3	4C	51	E7	4C	44<1D>	320
	2FF0: E7	20	1D	EB	C4	85	FØ	14 <fr< td=""><td>328</td></fr<>	328
	2FF8:84	B3	AØ	ED	A9	71	20	B5<77	32E
	3000: DF	1347	ZC	86	82	18	A5		32F
	3008:69	10	85	B-00	AØ	20	B1	90<57>	32F
	3010:05	B:01	FØ		C9	44	DØ	BA<1F>	330
	3010: A5	1917	C9	56	B6	04	Ç9	44 <d8></d8>	338
	3000: B0	22	38	68	A3	B/2		F7<61>	
	3028:E8	96	B2	66		18		A9KAE>	331
	3030:00	85	80	38	60	A9	CD	DO: 90.	323
	3038:04	49	20	FØ	02	A9	80	B5/FF	337
	3040: D2	20	10	ER	84	AC	20	53<45>	333
	3048: EA	150	28	20	28	E9	A5	B0176>	333
	3050:F0	98	A4	B2	B1	FR		7.0 /5	334
	3058:90	19	E6	F 2	20	53	C9		
	3040:F9	C9	20	90	04	C9	EA	90(50)	334
	3068:F1	C9	24	FR	Ø8		3A	98 <c4></c4>	335
	3070:00	717	18	38	68	24	D2	10 <p7></p7>	
	3070:FA	70	FB	CS	DØ	20	BI	F 3482>	338 338
	3080:09	28	00	07	CS	49		25<28>	337
	3088:02	85	DZ	A5	AC	85	F2	84<93>	337
	3090:AC	44	BS	A5	82	20	85	DF<17>	33E
	3098:00	DA	E.4	AC	FØ	41	20	ACKFB>	336
	30A0:DF	40	02	E9	18	AS	AC	ES(3E)	
	30A8:F2	85	F2	AB	A2	84	20	86<19>	339
	3090:25	A5	AF	85	DC	Δ4		88(8F)	339
	30BB: A6	AC	CA	BD	80	25	F2	88(85)	33A
	3000:91	97	CA	BD	88	05	88	18KCA>	334
	30C8:F7	AZ	08	AZ	88	20	86	25(93)	22E
	3000:56	91	70	9C	29	AØ	07	89 <bc></bc>	
	3000:02	00	91	97	88	10		98(01)	330
	301.0:48	AS	9D	48	A6	9E	AS	AF<41>	330
	30E8: 20	ED	E 5	B1	90	45	02	AB/90>	331
	30F0186	9E	68	85	90	CØ	86	68<4E>	33E
	30F81A8	BB	A3	24	02	52	24	30<86>	33E
	3100:02	C6	AC	A5	AC	85	F2	A5(BE)	
	3108:AF	1.0	07	A9	92	20	F7	ES(87)	33F
	3110:A5	AF	49	80		F7	Ea	18<96>	349
	3118:60	84	F2	BI	F3	C9	SF	F0<09>	349
	3120:F6	C9	41	90	27	C9		68(C3)	341
	3128:A4	AC	84	F2	28	60		FZKEDS	341
	3130:91	F3	38	E 9	32	90		C9<30>	342
	3138:0A	98	ØA	C9	11	98	ED	E9 <dc></dc>	342
	3140:07	C9	10	80	€7	AØ	24	86<1D>	343
	3148:04	26	05	88		FP	05	04<79>	343
	3150:85	D4	EA	F2	18	60	20	1D <e1></e1>	344
	3158:EP	84	AC	Bi	F3	C9	24	FØ<84>	344
	3160:07	20	CB	29	90	20	88	COKIE>	345
	31681E6	F2	20	10	EB	20	90	28<42>	345
	3170:20	68		80	B3	20		EA<84>	346
	3178:80	06	AS.	DS	29	F2		F5(F3)	346
	3180:20	DC	2B	A9	ØD.	20	A9	ØE<52>	347
	3109:20	F7	6.8	A4	94	A2	20	95<0C>	347
	3190:D4	99	00	21				Ø6<88>	348
		£5		94	18	60	20	10<75>	348
	31A0:EP	81	F3	C9	22		02	38<27>	349
		A9	ØF	20	F7		A5	94<88>	349
	3180:85	AB	20	F7	EB	E6		A4<71>	346
	31B8:F2	B1	F3	C9	9B			C9(EA)	346
	3100:22		EF	E.6	F2	CB	81		346
	3108:09	22	FØ	E 6	18	A5	94	E5<08>	34E
		A4	AB	99	88	21	18	68 <b0></b0>	340
	3108:AD	01	21	0A	80	24	45	94<08>	340
	31E0:C9	86	60	A4	FZ.	81	F3		340
	31E8: 7F	C9	20	DØ	23	C8	DØ		340
	31F@:84	F2	Bi	F3	C9	98		25<8F>	348
	31F8:29	7F	91	FS	C9	61	90	Ø8 <d2></d2>	346
	3200:09	7B	80	04	29	3F	91		346
	3208:08	C9	36	90	10	C9	3A	98(29)	34F
	3210:E1	C9	41	90	28	C9		90 <e4></e4>	350
	3718:09	C9	5F	FØ	D5	A4	F2	P6<92>	358
	3220:95	52	45	CD	95	44	41	54(ED)	351
	3228: C1	98		4E	50	55	D4	91(50)	351
	3230:43	4F	4C	4F	D2	9D	40	49\32>	352
	3238:53	D4	96	45	4E	54	45	D2<92>	352
		4C	45	D4	9B	49	Cá	90<00>	
	3240:9A			97		45		D4<6B>	353
	3248:46	4F	02 4F	54	4E CF	91	58		353
	3250:91		CF	91	47	4F	53	4F < DB > 55 < 53 >	354
	3258: 20	54 BD							354
	3260:02	BD PS	54	52	41	00	90	42<1E>	355
	3268:59	C5	90	43	4F		D4	9E <d6></d6>	353
	3270:43	4F	CD	9F	43	40	4F	53 <d6></d6>	356
	3278:C5	90		40	D2	90	44	45(E2)	356
	3280:07	9E	44		CD		45	4E<98>	357
	3288: C4	90	4E	45	D7	AØ	4F	50<2C>	357

90:45 CF 96 4C 4F 41 C4 96<AF > 98:53 41 56 C5 A2 53 54 41(RE) 4F 98:54 55 D3 A3 4E 54 ¢5<FB> AB1 CF 50 4F 49 4E D4 58<9B) 88:49 CF A4 4E CE 92.58 4F<7F3 88: 4B C5 AS 50 52 49 4번 D4<F13 02:90 52 41 99 41<FE CR: C4 AB 52 45 53 54 4F 52<8B> 45 54 DØ: C5 90 52 55 52 CE(9F) DB: A7 55 DE 53 4F<6ED 00:00 90 50 45 DØ AS RF 98(37) 91(50) 8:47 45 D4 AB 50 55 D4 0:47 52 41 50 48 49 43 D3(36) F8192 58 4C D4 4F(6C) 4F 92 50 77:50 49 54 49 48 CF 98 444073 28: 4F D3 92 44 52 54(67) 41 57 CF 93 53 45 54 43 4F 4C<C7> 18:45 02 60 AC AF 43 41 54< A00 20:C5 55 94 53 4F 4F C4 A6(87) 2B: 4C 50 52 49 4E D4 90 43(83) 30:53 41 56 **C**5 90 43 4C 4F<1B> 524433 38:41 C4 9A RR 88 R8 45 40:52 4F 20 44(99) 32 2D AA 92 4F 93 4F 49:50 4B 40 56<261 50: C5 93 2D 4D 4F 56 CS AA<503 58:24 C6 90 45 41<B1 52 50 45 50: D4 91 55 4E 54 CC 91(42) 4C C5 45 4C 45(96) 98(FA) 48:57 48 49 90 57 70:4E C4 90 53 CS 4E 78:45 44 49 **C6** CF 42 58KE73 90:55 D4 CF 47 45 D4 92<9D) 42 49 4C 4C 54 CF 89:46 90 44<653 98 4C 4F 4F DØ AB 90: CF 45(88) 96<93> 98:58 49 D4 DA 44 49 D2 AC. 43 CB 96 4C<DF 40:4C 53 43 CB ARL AF 96 52 45 4E 41(9A) 99: 4D CS 96 44 45 4C 45 54<1A> 91 E8: C5 58 41 55 53 C5 96(BA) CØ: 54 49 40 45 24 BD A7 50<43> AE. BB 45 58 45 18:57 C235 CISCERS 4E C3<C2> 90:90 45 44 58 4F 52 46 4F D2 08:91 43 4F 4C AAK SES FR: 24 CC 95 2D AD 97 52 ASCACO E8:4E 55 CD 44 92 45 CC BA(F2) 55 40 DO AA FØ: 44 54 52 41<980 FB: 43 CS DB 54 45 38 D4 96<862 88:42 4C 4F 41 C4 96 47 52(49) **08:** 55 CE 88 47 4F A3 A7 A34 76> 2A C2 92 56 49 10 I AA 41 4E(90) 18: D4 9F 43 4C D3 94 44 53<80> C4 DC 43 49 52<1B> 20:4F \$5 4E 28:43 40 C5 AB 25 50 35 D4<513 82<F2> 4F<09> 30:98 25 47 45 D4 9A 00 AC A4 38:80 BA BB 9B 47 \$Ø: 54 4E 53 55 C2 54(19) 48:CF 53 54 45 DØ 54 48 45< 6F > 50: CE A3 3C BD 3C BE 3E BD<AD> AA AB AD 58: BC BE BD DE AF(87) AF D4 50:4E 4F D2 41 4E E4KF7: BA 2BA A9 ED BD 3C BD 3C BE(41) BD BC BE BD AB AD AB<383 70: 3E 78:88 90 A8 AB AC 53 54 52(80) 9Ø1 A4 43 48 52 A4 53 D2<6A) 55 53 C3 45<000 BB: 41 56 41 CC 4C 41 44 D2 41 43<0B> 70: CE 54 CE \$2.28 45 CB 49(E1) 78: 4F 45 AW: CE 52 4E C4 46 52 C5 45( RF) 4F<07> DB 4C 4F C7 43 4C 48:58 30: C7 53 51 02 53 47 CE 41<643 88:42 D3 49 4E D4 58 41 44<D6> 4C C5 53 54 CP: 44 49 43 **CBC440** 28: 50 54 52 49 C7 53 54 52(08) 00:49 C7 44 50 45 45 CB A6<96> 081A1 49 AE 53 54 D2 A9 AFC9ES 014B 45 59 A4 45 58 4F D2(7C) B: 48 45 58 A4 44 45 C3 44<DD> 52 41 03 54 49(55) 0:49 D6 46 F8:40 45 A4 54 49 40 C5 4DCD30 C4 45 58 45 C3 52 4E<C63 20:4F 4E C4 54 35(5D) 09:C4 52 41 52 C3 25 80 25 BI B2(59) 10: 4E 25 18:25 B3 47 4F A3 55 49 AF (FA) 20:53 54 D2 45 52 D2 45 52(45) 20:CC 2C 8B 02 89<2B> 00 2B ac 35<673 50: BC 02 8A 9B 03 38:02 26 RF 36 02 28 03 B0(C9) 8D 02 81 02 84 03 40:02 88(83) 02 03 82 03 2B ØFKD75 48: 9C 8E 50:38 8C BF 20 02 12 @F<DD> 38:30 BC 02 PK 59 BA 02 6B<4C> 60:8A 02 60 02 66 02 67 02<5E> 68:68 02 69 02 6C 02 6D @2<9A> 70:B1 82 SE 82 82 **B2** 02<9F3 78:65 B2 02 44 82 02 D1 B3(F0)

3580:02 48 DF DØ. 3588:20 03 DØ BC 20 03 DØ 85<0E> 03 BS BR B5 3590:20 027E95 23 59 3598: 5F **B**7 02 83 35A0: D2 B2 03 85 R9 23 28 @F<B7> 3548:37 RC BE 20 82 0.3 10 DECRAS 3580:2F 1E 02 ØF 30 02 BEKR50 1F ØF<@A> 3588:31 02 20 ØF 32 02 21 83 3500:33 02 22 DUE: 34 CØ C6KACO 35CB: 90 03 BC 90 22 03 BD @E<45> 3500:2D 8C 90 02 B7 22 ØF 2E ( 8C ) 90 03 80(30) 3508: B5 22 ØF. **B**2 20 35E0:19 BC BC 1A 8C 99 83 M2<145 97 AF(6E) 35E8:03 C4 12 03 C4 12 35F8:90 83 BE DB 82 90 98(24) 35F8:02 91 02 1C 88 03 09 RECORD 3600:06 70 03 BE < 49> 14 02 16 03 3608:98 82 C8 CB 03 10 SC(B4) 3610:03 92 02 B5 03 PF 824A7 06 12 03 02 03 80 50:8167 12 BE < Z10 3628:12 12 90 85 23 90(AB) BE 3628:02 90 03 96 02 90 03 BE<85> 5630:0B 92 BX RE DR 002 85 DBSERS \*6\*8:82 03 C1 91 92 C1 92 BZKE10 3640:C1 98 8C 12 02 96 02 90<75) 564R: 03 85 12 02 03 BE 824765 12 97 83 8C 12 8C 83 3850:12 984100 3658:02 ac. 12 12 **C4** 90 BC Ø35A9> 3660:C8 D7 70 83 BC 03(44) C5 90 3668:86 03 83 12 02 92 93 17(04) 3670:06 82 18 C6 02 62 C2 Ø2<8A> 3678:6A 83 包色 C7 23 12 C6<4F3 BC 3680:02 03 82 28 0F 79 SECAS) 28 ØF 20(16) 3688:20 Ø2 85 3B BC 3690:03 CB 07 02 03 Ċ9 **M2KSF** 12 98(DC) 3698:03 8C 19 CA 98 82 8C 36A0:03 B1 02 84 AE 02 CD AD<223 3648:02 03 AE **02 03** AB CC @3(19) 3680:8C 02 25 03 CD AD 02 03<ED> 92(F2) NARSING OF MY CD MY MX 12 3600:15 83 48 82 41 82 42 02<2E) 30 02 3608:43 92 50 203 35 02460 3600:58 83 80 04 03 12 RF 30(33) 3608:03 82 83 28 BE 36 P3 REC5A> 36E0:12 92 03 80 00 DE CDA 95 12 87 36E8:3C 85 SF 20 Ø3 90(07) 36F0:03 26 90 02 25 98 82 984 955 36F8:03 02 11<A9> 9C 12 93 93 03 3700: E9 EΑ 72 E9 97 EA 3708:E9 76 E9 6E E9 11 EB F2<9F> 37101F0 AE EE A3<83> 78 EE 80 EE 37181EE EE EE BE EE 30KCA 3720:EF 83 EF 96 EF 94 EF 98(45) 3728: EF 10 FB SC EF 2D EF 37(19) 3730: EF 38 EF D2 EF 14(17) 06 EF 3738: CF 73 EF 99 EF 5F 58(80) 3740: FF 55 EE 28 EF 86 90<545 EE 3748: EF 48 EF 44 EF 26 FØ 30<74> 3750: EF 64 EF 2A FO FF EE 22(08) 63 3758: EF ES EF EF E9 EF 95(01) 3740: EE BE EE C3 EE C7 EE CB(A0) 3768: EE CF EE DY EE DC EE 27<02> 3770:FØ DF EE I.D. FØ E7 EE 1E(33) 3778: FF 32 FF 47 FF 40 FF ARCOSS 3780:EF B2 EF AL EF A4 EF BC<2E> 84 3798: EF AB EF BB< AØ> 3790:FF CA FF DA FF DD FF FD(1A) 14<23> F4 3798: EF FI EF EF 18 F@ 37A0:FB FB EF 85 FB ØB FØ ØE<80> S2 EF CE EF 7E EF 26 EF F7<30>32<86> 37A8: F0 4E EF EF 3780:EF 60 3788:FB 20 28 60 A5 B9 C9 3700:90 10 09 28 FØ 18 09 28(B5) 3708:98 16 C9 56 FØ 12 09 57(84) 5A FØ ØA C9 5D(14) 3700:F0 0E C9 3708: FØ Ø6 C9 61 F0 02 38 60(46) 37E0:40 60 E9 26 1E F4 A0 20 60> 37E8:84 A0 84 A1 84 B9 88 84<44> 62(98) 7F 37F0: AD A9 BS AE BD FE A7 (6A) 37F8: 20 94 94 AB ca 04 3800:80 2E AS A8 48 20 00 E4489 85 AB A5 D2 10 06 20(97) 3828: 48 3810: BE C2 4C 35 F.1 20 FB E4 (0F A1 85 A0 A6 A8 E4 A7< 08) 3818:84 A7 FB(23) 3820:F0 BA AØ 80 E8 **E4** 3828:03 20 FB E4 85 AD 84 AE'5D 30136 3830:20 62 C9 AØ Ø1 B1 88 3878:34 AE 98 ØB DB 20 BBKSE. 3840:81 8A C5 AD 70 02 D0 254682 3848:AD 1F DØ C9 Ø7 FØ Ø6 A5<16:2 3850:14 29 MF DB F3 28 87 F2<88>

Listing »Turbo-Basic XL« (Fortsetzung)

						_		
3858: A5	11	FØ	WE	AØ	92	B1	BAKJES	
3860:18	65	SA.	85	84	92	CC	E6KCD/	
386B: 8B	82	CB	63	11	A5	B5	FØ4E5>	
3870:07	20	FB	E4	A9	20	35	BS <ac></ac>	
3878: 8D	FE	02	20	53	F7	4C	2B< 1E >	
3880:F8	86	AA	85	96	84	95	A4.69 ·	
3888: AA	A5	AF	FØ	10	63	AF	B1 <b6></b6>	
3890:95	30	03	CS	DØ	59	CB	20(07>	
3898: 07	F1	4C	Ci	F1	18	98	65: 41	
38AØ: 95	85	95	90	02	E6	96	60 FA)	
38A9: AØ	FF	84	AF	26	AF	A4	AF (833)	
3880: B1	95	48	C9	9B	FØ	0.4	29 59 /	
3898: 7F	FØ	03	20	96	CZ	68	10(08)	
3800: EB	60	20	82	F2	20	£3	F1<86>	
38CB: A9	70	4C	96	C2	AØ	90	B1(B5)	
3800:8A	AA	CB	B1	AB	20	52	F9<1E/	
3808: AE	91	F3	F 2	11	A6	69	AØ<05>	
		8A	20	9A		FØ	24 65	
386.0:04	B1				F 3			
38EB: C9	40	DØ	02	CA	CA	86	D4 <fd <="" td=""><td></td></fd>	
38FØ: 20	07	F2	C9	D4	10	F9	AB <fc></fc>	
38FB: Ø2	Et.	BA	85	9F	CB	B1	BAKEZ>	
3900:85	A7	C8	84	AB	20	49	F2(E1)	
3908:44	A7	C4	9F	90	FØ	60	20-80>	
3910:3A	F3	C9	36	FØ	35	C9	54 06/	
							-	
3918:F@	1,7	20	46	F3	20	TA.	F3<20>	
3920:09	37	FØ	84	C9	02	82	20-19	
3928:20	38	F-3	20	96	CZ	4C	624CF > -	
3930:F2	AB	1E	AD	91	F3	DØ	22 BC /	
3939: AD	82	84	94	A9	20	28	96(01)	
3940:C2	C6	D4	02	F7	4C	94	E246E +	
3948:20	3B	F3	0.0	04	20	28	F3<38>	
3950:20	18	ØF	49	80	20	B1	F3-10-	
3758:C9	A8	DØ	EC	22	39	F-3	4C <e5></e5>	
3960:82	F2	C9	8F	FØ	2E	E	55 78	
3960:48	20	60	E4	C6	AB	68	C9<7D/	
39/0:00	DØ	12	A9	24	28	96	C2 <bb></bb>	
3978: 20	25	33	20	05	DA	09	88 90>	
3980:99	78	02	DQ	200	28	BE	2B <bb></bb>	
3988:20	50	F9	40	82	F2	20	E2<78>	
3990:F1	4C	87	F2	20	38	FJ	85×8A /	
3998: AF	A9	72	28	96	C2	A5	AF < 08 -	
3940:FQ	13	20	38	F3	C9	22	DØ<25>	
39A8: 05	20	96	C2	A9	22	50	96<37>	
3980:C2	C6	AF	DØ	ED	A9	22	20< F5>	
T988:96	C2	4C	87	F2	38	E4	18<09>	
3900:85	AF	A2	20	A9	ED	AB	71<9E>	
3908:20	89	FI	20	3A	F3	C9	5A(E2)	
3900:F@	23	C9	SD	F0	15	C9	61 (RØ)	
3908:F0	1 E	C9	62	FØ	17	C9	6A<25>	
39E0:F0	13	C9	28	FB	12	C7	3D<41>	
3966:80	A4	AØ	20	B1	95	29	7F <bc></bc>	
39FØ: 20	57	EA	PO	99	20	92	F2<83>	
39F8: 2Ø	EE	F1	4C	82	F2	E6	AB(64)	
3AØØ: A4	AB	C4	A7	BØ	83	81	8A(CB)	
2988: 90	68	68	60	48	20	59	F3<7F>	
3A10:68	05	AF	AZ	01	A9	ER	ARKEAS	
3A18:5A	20	ÐĐ	F1	4C	FF	F1	C9<11>	
5A20:08	FØ	1F	09	30	FØ	LB	C91 A5>	
3A28: 3E	FD	17	C9	45	FØ		C9 <fb></fb>	
						13		
3A30:4F	FØ	ØF	C9	07	FØ	12	20 F2>	
3AT8: 9A	F3	DØ	ØC.	C6	B9	C6	B9 <cf></cf>	
3A40: 10	86	E.6	89	63	89	36	F4 <eg></eg>	
3A48: 68	AS	A8	48	20	66	35	68 <f4></f4>	
3A50:85	AB	EB	18	DØ	EC	60	01(CE)	
3A58: B1	88	49	26	BD	91	F3	60(07)	
3A60: 09	89	FØ		C9	3D	FR	@E (C3)	
			12					
3A68: C7	3F	F 20	OA.	C9	46	FØ	26 <f4></f4>	
3A70: C9	51	FB	02	C9	41	60	B5<4C>	
3A78: AF	A2	20	A5	83	94	82	26 <fc></fc>	
3A98: BB	FI	4C	€2	F 1	84	B3	B1 <e6></e6>	
3A88: BA	DO	23	CB	81	8A	49	80(95)	
3A70:85	C7	20	88	E4	A6	90	86<19>	
3A9B: C4	A6	91		C5	20	03	F7(85)	
			86					
3AA0: 80	SC	DØ	20A	AB	8C	BI	904.70	
3AAB: 05	C7	DØ	F 1	E.Q	68	A5	C4<66>	
3AB0:85	92	A5	C5	85	91	A9	00 .9A>	
3AB8: 20	54	F6	20	68	E5	AR	00 <f1></f1>	
3AC0: 20	80	F6	20	BB	2F	A6	AB(AB)	
JACB: EB	E4	A7	50	23	20	6P	E516B>	
3ADØ: AØ	06	20	80	F6	AD	07	F7 <f8></f8>	
SADB: DØ	2C	A5	C7	BA	ØC.	91	C4<5A>	
3AE0: A9	20	FØ	05	40	64	AB	84(A7)	
3AE8: B3	48	A9	04	20	54	Fá	68 <a3></a3>	
3AF Ø: AØ	00	91	C4	A5	88	C8	91/F6>	
3AF8: C4	AS	88	C8	91	C4	A6	B3<17>	
3800:CA	8A	CB	91	C4	60	A5	90(80)	
3B08:48	A5	91	48	AS	C4	85	90(09)	
3810:A5	C5	85	91	A5	D4	48	A5 <ee></ee>	
3B18:C7	20	83	E4	68	20	E5	F4<98>	
3820:90	68	58	95	91	68	85	90 <f3></f3>	
3828:80	BØ	68	68	A9	<b>6</b> 8	A2	<b>09</b> <18>	
	Apr are							
\$B30:20	<b>Ø</b> B	F7	91	BA	DØ	83	C8<@4>	
	88 88	F7 49	91 80	45	C7	603 F@	C5 <c5></c5>	
\$B30:20								

3848: AS RE 85 BA AS RE 85 88(98) 3850:60 B1 8A D0 03 C8 B1 8A< 94> 3858:49 80 85 C7 20 D3 F7 BRK 3AS 98 C5<F9> 3BAD: E2 DO ED AD ØC. B1 3848:07 1347 90 48(F8) AØ 86 Frt. 3870:85 E0 C8 B1 98 85 C8<61> 3878:81 90 85 E2 CB Ta 1 90 RSKEND 3888:E3 CB B1 98 85 E4 CB B1<75> 85 **A5** 20 B.S 3988: 98 E2 C7 3890: 20 9D B-2 29 CØ E5<69> 3898:68 20 ES F4 98 48 49 11<05> 24 F5 A9 28 4C 2B 3BA0: 20 F8<16> 2E E5 EC 60 22 38A8: 4C 85 R1<47> 904745 3880:90 D4 DØ 2A CB B1 3888:C5 D5 D0 LC CB B1 90 CSKRAS 2BC0: D6 D0 15 CR R1 90 C5 07<505 CB C5 D8 DØKCC) 3808: DØ ØE B1 98 86(8A) 3800:07 C8 B1 90 C5. DΩ FØ 3808: 6A 45 EC 45 D4 88 49 Ø5<82> 3BE0: D4 45 EC F9 49 88KBA 10 6A JBE8: 0A 60 18 65 90 85 90 85(FB) 90 91 84 E6 BF TREE: DE E6 68(D2) 3BF8:CB C4 BØ A7 83 213 DE C2<49> 3C00: AS 88 85 8A A5 89 85 8B(11) 3008:A9 00 8D @7 F7 BD A3 F5(7B) 3010:AB 03 BB H1 BA 3C18: 9F 88 B1 BA 30 15 20 E7 (0E) 3020: FA 20 AF F6 .29 9E F6 49(DD) 3028:00 85 87 85 98 85 B6 60(A4) 20 00:3A 30 10.20 C4 F5 4C E6 60 SCAR-F4 AA AA BD 21 FR RAKRAS 3C40: A6 AB EB E4 PB 19 87 40 683 3C48:9D FF C9(AA) A4 AB 88 BI BA 87 3050:1B F0 29 A9 41 ARKEE A2 9F 3038:40 4C 0A F7 A5 B5 A7KC6> 49 A3 3C60:68 B1 BA 26 BD F5<5A> A3<12> 3068:68 88 A9 88 AC. 3070±F5 F0 03 40 00 F8 20 04\37> 3078±F5 20 94 02 A9 08 85 95(40) A9 40(4A) 3C80: A9 85 96 210 AG 61 3008:2E F9 91 SA 30 07(74) 3E90:85 BB 88 B1 80 EE5 RA 49:375 3098220 85 85 85 53 54<585 94 68 30A0:4F 50 50 AR AD 01<CB> 45 44 3CAB: B1 8A 10 EB A5 BA 85 ABCCSS CDD: A5 BB 20 AB . 29 > 85 A1 62 C9 85 9F 40(84) 3CB8:07 61 BA 6B 68 CC0:D7 FF 20 3A CA 85 BC B4<64> 3CCB: BD 60 84 83 20 AB E5 28(E6) 05 3CDØ: 25 BØ 10 A5 DØ 18/ B3> 3CD8:A5 D4 FØ 14 85 89 04 ASKCDO 3CE0:88 B1 8A 48 C6 89 FR 89-10> Fa 3CE8: 20 68 35 EB 12 F5 68(A1) C9 FØ SCF 01 60 68 6Z SCERVED ID 48 20 FB E4 68 C9(BB) 3000:17 FO 05 A9 1E 20 23 F4<68> D09: 45 D4 A4 05 4C 3D10:51 20 23 F4 3D10:6F F8 85 64 40 85 FB 4C/ BB> 18 A5 70 85(50) 91 D20:C4 65 AB A5 85 C5< 30> 64 3028:49 00 CD E6 02 90 89 DØ 8F> 3039: 18 CC 65 02 90 02 DB 09(97) 91 90 BE (92) 3D38:85 85 ØF 84 84 D40:68 A9 82 4C DC FB A5 D4(F6) 5D48:91 C4 C8 A5 05 91 C4 CB<CB> 91 91(FB) 3D50: A5 D6 CA CB AS D7 3058:C4 C8 A5 D8 91 C4 C8 A5(A9) 3060:09 91 C4 60 **A5** BC 85 BEKIC> 3D68:85 90 85 0E A5 8D 85 8F492> 3D70:05 91 85 0F A6 86 B6<FF> 60 3078:F5 A4 84 F6 A6 F5 E4<83> 3080:88 A5 F6 **£5 89 80 2**3 ARKER> F5 3D88:00 B1 29 FC 91 F5 A8(48) 3090:02 AZ 06 9 90 91 F5 C8<83> 3098: CA DO FA A5 F5 18 69 08<D7> 30A0:85 F5 A5 F6 69 88 85 F6(58) 3DAB: DØ D3 48 84<FD> 4C C5 C9 22 30F0: BA 84 BB 84 B9 B4 84(29) 3D88: B6 84 B7 **B4** 98 68 84 RD(FD) 34 49(94) 30C0:B4 11 4C **C**5 B1 8A 8D 07 FF(4C> 3DC8: 26 F7 60 90 AØ F7 8D 34 F7 8C 80 85 D7 A5 8A 38(93) 30 3DOM: SE 3DD8:F7 A9 85<69> DEM: BE AS BR 85 94 A7 C4<C3. BF SDE 8: 9F 19/2 30 84 95 C8(B1) 3DF0: 81 SA C8 84 AB C9 00 F0<07> 12 C9 00 IDFS: ND C9 20 FØ DØ<Ab> SE 00: E 4 ØG EØ 68 A5 D7<20> C9<77> SE08: F0 FR C6 D7 4C 1F F7 3E10:07 D0 09 A4 A7 88 B1 86<82> SE18: C9 1B FØ C9 E6 D7 C5<99> DØ 3E.20:20 82 F4 A9 4C DC FB(A7) 16 3528: AU 01 B1 BA 30 F2 18 A5(R2)

E40:81 8A 85 9F CB 84 A7 40<10> 3E48: 1F 28 1F F4 20 00 E4<F8> 38 59: A6 AA 21 00 27 BD 60 20(58) SE 58: 03 Α9 3E A2 20 03 F7 80<98> JE 60: F 7 00 E4 ZE58: 28 0.9 KC DØ BA AA AA BC< 15> 28 24(59) 3E 70: 60 21 DØ 8A A9 04 A9 3E78: E5 4C 28 EA 60 20<0F3 30 F7 BØ. 98<85> 3880: D3 1.0 C9 3E DØ 3E88:20 2B F8 A4 B2 88 84 A7<7D> TE 90: 60 A9 2C A9 4C DC<245 17 18 TE 98: F8 A5 C5 91 BA ASK RET 0F 98 37 A5 90(F3> VEAR: 8E C5 98 89 38 SEA8: E9 04 85 90 85 ØE. 80 04(FA) VER0: Co 91 C6 9F 40 03 B1 98 195 VER8:85 B2 88 B1 70 85 A1 88<ED> 35 CO: B1 90: 85 AØ 88 B1 70 DB< SB > E9 00 8540F> 3EC8: 11 BA 38 **A5** 90 T DE: 98 85 DE BO 24 CA 91 C6K502 TEDS: OF 98 20 D3(21) 1 F 62 38 60 TE0:F7 Bb 2D C9 ac FØ ØA 09(F3) 50 DØ FF F8K5C> TEE8: 1E ED 26 C9 82 D1 12 85 1A ASK5D> MEFR: 1F Ad AR DR FE 85(10) SP AS AB SEFB: A1 T- 20: 8A D: 85 A7 AB 02<38: TE 08: 81 8A 85 9F 68 A9 RF 20/50> F8<21> 2C F 7 A9 1A 4C DC ZF 10: A9 120 F6 3F 18: 2Ø 0.3 BØ 22 A9(D1) DØ TFT0:08 AA E8 4C 08 F7 A9 1E(5A) SE 28: 00 C7 20 D3 F7 B(2) 14 C9KDB> C9 00> 3F 50: 50 FØ RE C9 51 E.F FØ 3F 78: 1E Ø8 C9 FØ AC DO ER 384000 2C SEAD: E9 A4 49 AC DC FR. DE S 1C 3F 4B: 20 F4 A2 A4 A8 B1(BF) 15 CI 3650:8A D0 93 CB BI BB(22) BA 3F58:20 ED E5 03 9D DØ DEKEED JEAD: AD 03 B1 9D 85 88 88 B1(6D) JF 68: 90 85 8A 91 BA 9F CBKFE> 85 3F 70: B1 85 60 19 4C DC<47> 18 2C A9 19 4C 03 F7 BW F6 C9 3F78: D4 89 SERØLER. 20 45< DA> 3F88:08 A9 04 24 A9(6A) 20 3F90:45 4C 2B FB 4C 50 F7 A9< 03> TF 9R: 15 20 49 2.73 20 A9 17 20(44) 49 3FA@: A9 85 B9 11 20 DB 8D(D5) 3FAB: FC 02 F5 A4 26 C4 BD 38KE7. 3F B0: 11 AS BC A2 80 86 80 A6(AF) 3FBB: 89 B6 C3 A2 98 86 89 ACK TAX FF 03<EA> 3F C@: AØ A5 89 C9 80 00 F5 37(2E) SF CB: 4C B0 20 94 C2 A9 A6 B9 3FDØ: 20 46 F3 A9 DB 20(39) 3F D8: 52 F9 A5 B9 C9 1F 90 22<03> SFED: E9 85 AF 4C **OCKEB**2 JFE8: AZ 00 A9 F9 A0 67 20 BB(71) SEED: F1 28 FC F1 A8 81 B1 BACAR 49 3F F8: 30 15 67 AD F9 601 5F > 20 4000:F9 AQ 01 B1 SA 85 D5 88< 20> 4008: B1 8A 85 D4 20 FQ TRY 1R 56 4010:94 C2 20 66 FF 4C 7B E6SAR 4018:85 D5 86 D4 20 DC 28 70<75> 4020: BE 2B F3 A4 FA 95(F9) A5 85 94 4028: B4 40 F2 F1 20 41 34/04 4030:20 4E 49 4E 45 A0 3F 42. 3A 41 C4 4D 56<7E2 4038:40 4F 45 CD 4040:41 4C 55 C5 3F 23 56 41KD5> 4048:52 D3 24 4C 45 CE ZF. 44 ( DB) 4050:41 C1 3E 36<55> 4058:87 49 45 50 55 D4 44 49 E 0 4868: CD 53 54 41 43 CR 4F 56 062 52 46 4C 4F 4060:45 SE 4CKAD> D7 4070:49 4E C5 SF 46 45 D2 54(98) SEK SAN ARTR: AF AF 20 4C 4F 4E C7 53 55(3F) 47 4F 4080:44 45 CE 3F 4088: C2 47 41 52 42 41 47 C5(DI) 4090: 3F 43 48 DZ 4D 45 CD ATKIA 4098: 3F 40 4F 41 C4 3F 45 45 8R 40A0:53 D4 3F 48 49 CS(FF) 57 4C 40AB: 3F 50 52 45 45 41 **D4** 4080:44 CF 3F 45 58 49 D4 56<18 4F 42B8:50 52 3F 45 58 45(OF) C3 4Ø08: C3 50 4008:47 53 20 4F 50 45 CE SFKEE> 4000:44 45 D6 57 52 20 4F 4E< C3 > 54<2B> 4008: 4C 43 4D C4 4E 4F 50 45 52 44 4000:28 4F A3 CE 40FR: 20 4F 4F 4C DO 45 4F DA< DB> 40F0:54 52 4E C3 54 55 49 40<83> 40F8:45 4F 55 D4 4E 41 CB 21(7C) 4100:46 52 41 4D C5 4.3 55 4F 41@B:53 D2 21 4F 56 45 524.02 4110:52 55 43 48 4R 55/21 CE 53 4118:CD 44 4F 44<12> AF AF C5 4D 4120:C5 4E 4F 54 20 49 4D 50.AE.

3E30:9F 65 8A 85 8A 90 02 E6<93>



4128:CC 52 41 CD A0 A0 A0 A0<7C 4130: AR AR AM AØ AØ ΔØ ΔD ARCIDS 4138:44 31-BA 3E 23 46 49 4CKCED 4140:45 D3 44 53 48 20 46 55< 6€ > 46 414B:4C CC 41 54 41 40 28<80.4 47 CF 46 4150:49 2F 4C 45 A3KF60 4158:4E 41 4D C5 50 4F 49 4EKSER 4160: D4 4C 4F 43 4B 45 **C4** 44<10> 3F < E5 > 4168:43 4D C4 3E 44 49 D2 4170146 49 4C C5 50 4F 49 4E . 64) 4178:D4 3E 41 50 50 4E **C4** 42<57) 20 4190:41 44 53 45 4.3 54 4F< 170 05 4188:52 B3 14 04 13 AA 12(04) 4190: C5 14 DØ F6 46 BA D5 FI4 < DD > 4198: D4 DC 2B A2 20 25 BD ARC44 > 95 41A0:FB E.S CA 1.73 FB 20 9F.AA> 41A8:26 20 9C 2E 68 85 D4 A9<21> 4180:00 85 05 20 DC 29 9D4.5B2 20 4188:20 49 69 85 D2 85 D3 467.995 A2(89) 41CØ:3A E5 20 C4 FA C6 98 4108165 BD 6E FB 95 EØ CA 10(69) 28 4100:F8 20 70 AR 0907 85 D4<FR 40 41D8:C9 00 02 84 D5 84 F2(1B) 41E0: A9 24 39 FB 28 A9 60 20 (01) 41EB:30 FB A9 50 20 30 FB AB(53) 41F0:06 **A9** 80 4C 6F DA 48 20(D7> 41F8: B8 20 28 90 2E 20 BB 2F<FD> 4200:68 85 D5 20 9E 26 A9 88(4D) 4208: A4 **D4** CØ 40 DØ 85 **A4** D5<80> 4210:85 D5 98 AA 40 48 40 46<922 421B: 20 58 FB BA 29 ØF 0.9 304 AE A228: 09 34 学品 02 69 86 84 F2<46> 08 4228:99 RA E6 F2 58 41 @2<F10 88 43 4230:56 00 20 84 32 **DUDICIAA** S 28 423B: 00 (2)(2) 00 E4 20 73 C5<FØ> NO. 84 F2 94 4740 z AD F1 RA FZ(RR) C9 18 424B: 20 **C4** FB 80 35 85(28) 4250: ED 20 FC 64 13 20 FB C9(BD) 4258: 3C 20 29 BØ 29 DE FB 13(3D) 20 C4 C9 4268: FC FB 30 RØ 10(07) 4268:20 DE FB 20 F2 FB 20 EC<7D> 4270:FB **A5** EØ **A4** E1 **A6** E2 85< BC / DUKEE 4278:14 84 13 86 12 C5 14 93 4280:F6 40 73 C5 20 C5 4C< 130 4288:07 FB 20 D2 2B E6 F2 BØ< 950 4290:F3 BA 85 E3 ØA BA 65 E3<@3> 20 D2 4298:85 E 3 **2B** E6 F2 BØ<30> 42A0:E3 65 **E.3** 60 18 65 EØ 85 ( DC ) 70 06 42A8: EØ E6 E1 DØ 02 E6<5F3 60 86 E.O 26 26 42BØ: E2 E 1 EZKDE 2 4288: A4 E2 AS EØ. A6 Ė 1 86 EØ<DD> 26 42C0:26 E1 26 €2 EØ 26 E1<4F3 88 65<88> 4208:26 E2 65 EØ. 85 EØ 42DØ:F1 85 E 1 98 45 F 7 85 E2<6B> 4ZD8: 6Ø 20. EC FR A4 E2 A5 EØ<17> A2EM: AA F 1 014 EØ 26 E1 26 E2(21) 42F8: AS EØ 85 E DI RA. 65 E1 85<400 98 42F@: E1 65 E2 86 EØ 26 E1(72) 85 E2 AØ 60 ØIC. 42FB: 2A 20 F5<215 99 F8 4300: B5 DB 20 DO F4 AA(99) 4308:AA BD 00 21 **@**3 28 34<26> 30

4310: DA 20 DC E4 A5 D6 25 D7(A2) 431 R: FR F2 AR RR RA DR PLA DE< 77> 4320: B1 D4 1.8 62 C6 DB 20 DØ< 481> 4328: FC DIA. ØA 85 A9 A2 20 20(00) 4330:06 42 2A F4 AD 82 A3(91) 85 4338:20 18 FD BØ BE AB SH F14< (0F1) 434Ø: DD **A4** DC B1 A2 45 DB 85< 81> 4348: DA 86 FD **A4 F**3 RA no B1(A2) 4350: DE 3D CØ 05 90 63 1D CBKEAS 4358:05 71 DE CA DD FØ 27 28(82) 4360:0B FF **C4** E1 98 E7 28 E0.E35 DC 4368: FE E6 A5 DC C9 BØ<03> 4370:06 99 65 CS ED 90 A5\E4> 4378: 9B 69 07 85 919 99 62 EAKE@> E6 D4 DØ 02 E6 4380: 9C D5 A5<38> 438B: 06 DØ **B**2 C6 D7 C6 D6 4C(D3) 4390:4E EC. 400 20 68 PHP1 AB 8A.7A> 4398: ØA 20 24 29 63 AA 98 5D(D2) 4340+CC FC 68 18 19 AC: 18 30 (F9) 43A8: 38 AB 60 CO CB CB CB 18KDE> 43B0:00 CB CB 28 1.4 14 PA. 80(85) 28 43B8: 14 14 2FI 28 28 2自 2842E3 4300:28 14 28 00 200 60 82 MEL ARS 83 82 4308:02 03 01 01 01 ØØ<BE> 43D0:00 03 82 88 81 83 87 BB(79) 4308:F@ FC FE 84 ARC 393 82 (EL) 38 43E0: A5 57 29 BE AA A9 88 85(D9) 4 SERL DE A5 99 MA 26 DE ØΔ 26(05) 43F0: DF 99 70 82 Eò DF BA<CA) 43F8:85 DE 26 DF FC 85<91> BD DD 4400:E0 90 C5 99 FØ 90 D7 BC<CB> 4408:FD FC 84 E2 BD<4D> **B**9 11 FD FC 4410: CO 25 RO ED 65 E1 48(A2) AA FØ DE(81) 4418:4A 40 AA 87 86 4420:26 DE CA DØ F9 18 **A5** DE (F9) 4428:65 58 85 DE A5 DF 59(AC) 65 4436:85 DE 45 90 85 E4 **A5** 9B(21) 4438185 E3 84 E2 84 ED FØ **聞CKSA>** 66.42> 4440:39 PD ED 85 FD 46 E4 444R+FX RR. DØ E9 45 E4 DØ GEZERN 4450: A5 E3 E5 E1 RØ. 8B AA F74 795 **A5** 4458: DØ BA EC. CB 20 DØ ADCA1 > 4460: CB 05 18 ARI BC. AD ED R4< 07> 4468: E4 BD 14 FD 85 EE **A5** CB(53) 4470: DD CØ 05 4D CB 25 99 CB(FB) 4478:85 EE **A6** BA CA DØ FC BB(98) E4 4480:10 F4 **A4** AD CZ 85 99<D7> 44BB: CØ 85 A6 38 EE 2A CA DØ<13> 4470: FB 88 18 F3 18 68 20 BC< 775 4498: E5 85 DØ FB 28 14 FD<71> 44A@: BØ F3 A5 99 85 A2 **A5** 91(27) 44AB: 85 A3 AD E5 02 E9 96 R5< 0.40 44B@:E7 AD E6 02 E9 80 85 E8KEC> 44BB: 18 **A5** A2 69 03 85 **A2** 90<73> 4400:02 E6 A3 C5 E7 A5 A3 E5<773 44C8: E.B. 90 83 4C 7B F6 A6 ED<71> 4400:A4 E3 20 26 FF FØ 83 4C<53> FF(F1) 44D8: BA FE 28 18 FF 20 14 44EB: 98 30 0h 20 26 FF DØ Ø6<36> 44FR: 20 18 FF AC 1.7 FF 29 **QB**<67> 44FB:FF 98 AØ 88 91 **A2** BA @A<@3>

44F8: 0A CB BA 91 **AZ** E3 A6<293 4500 ED 20 MB FF C4 E1 BØ ØB(6A) FFKEE) 45009:20 26 FF DØ. **BA** 20 1 B ABKAD> 451D:40 3R FF 20 14 FF 98 91 451B: 02 A2 80 RB 1.1 **A2** 917/093 4520: A2 04 99 CB CA FØ RO TRAN DE FE 20 4528:28 FA FF AØ. Ø1 (4A) 453Ø: B1 29 87 C5 **A2** ED CB B14535 4538: AZ **E**5 **E**3 90 OR AR Ø0 R1(49) 4548: AZ 91 09 80 **A2** 4C F2 FD(F1) 4548: 20 EC A4 99 88 C4 FE EØ<263 4550:80 28 20 ED FE 20 FA FEK DED 4559: AØ 21 B1 A2 29 07 C5 ED<FF> 4560:C8 **B**1 A2 E5 E3 90 N8 ABKC7> 4568:00 B1 A2 29 7F 12 **D4** 20(E3) 4578: DE FE AØ 61 B1 A2 29 07<F23 4578: AA CB 81 **A2** AB 20 ØB FF(74) 4580:20 ØB EF 86 ED 84 E3 38< BC3 458B1 A5 A2 E9 03 85 82 BØ @2<AE> 4590: C6 A3 C5 90 DØ 86 **A5** A3<500 **B1** 4598:C5 91 FØ 15 AØ 20 A2<40> A5<98> 4500: 10 BA 30 BA E6 99 18 450R DF AS. E1 **65** DE 90 92 EACRAD 4580: DE 68 **C6** 99 38 A5 DE F5< DAD 4588: F1 85 DE BØ 02 C<sub>6</sub> DE 60<CF> 42 29 45C0: A0 80 R1 **7F** 85 FECROS 40 45C9: C8 **B**1 **A2** 48 40 85 ED(CE) 98 45DØ: 68 E4 E4 E8 **P**335 A2 @BKFR5 45D8:C8 60 CA 1 2 03 A6 E4 88(9B) 45E0: 60 81 DE 3D CB Ø5 1 D CBKEBX 45E8: 05 91 DE 69 **B**1 DE 1D CB<D4> 5D 45F@: @5 CØ 05 FØ 24 AD CS(ØD) A9< D6> 45FB: 05 60 AD CB 05 FR **0**3 4600:00 68 A9 @1 68 20 45 DE CDAD FE 02 460B: AD 48 8E FE 02 20<3B> 4610:9C C4 7D 9C AØ 20 C2 AB(EE) 4618:8D FE 82 4C 49 **C2 B**1 BA(29) 89 89 4620: C9 26 FB 4C A2 76<50> FF DØ A9 4628: AØ 06 AZ A2 M2<CD) 46 30: AB B1 AF FF 80 BE 88 FF< 06 4639:9C B1 FF 60 A9 SB 20 96(65) 4640:CZ AØ 88 91 8A AA CB B1<E@> 4648: RA 28 52 EO 40 5D 20 96<BC> 4650:C2 API **P**2 R1 88 4C **B**3 FEKOD) 4658:20 87 FA A9 ØC. 4C DC FB<AF> 466R: 28 15 F4 28 FB **E4** 85 AØ< D7 > 4668:84 A1 20 42 69 19/3 EQ 6B(2F) 4670:68 A5 11 FØ 15 AZ **M**2 B1<1F> 4678:BA 85 SE CS C4 QE Pi2 1 D < 3.4 > 4690: B1 88 85 A7 CB B1 BA CB< 425 466B: 84 AB 20 DM **A7** FIF **A4** ASCRE > 4698:11 DØ E9 AC. Δ4 E5 PΙΔ 9D< 14 a 469B: D5 6C 99 22 FF AØ 61 B1(22) 30 12 **A5** 9F 46AØ: BA 18 65 8A(2D) 02 46AB: 85 AB. 90 88 E6 81 BAK BID 46B0:10 C3 4C 78 6A F5 40 E6<05> Laenge 18104 Bytes

Listing »Turbo-Basic XL« (Schluß)







# Schnell, schneller, Turbo-Basic XL aus dem Compiler

Passend zum Interpreter für Turbo-Basic XL gibt es jetzt noch das Turbo-Basic XL als Compiler zum Abtippen. Er macht langsame Basic- und Turbo-Basic-XL-Programme superschnell.

mmer dann, wenn es bei Basic-Programmen auf die Geschwindigkeit ankommt, ist eine optimale Programmierung erforderlich. Man sollte unnötige Sprünge vermeiden und weniger benutzte Programmteile ans Ende verbannen. Dies ist aber nur als goldene Regel gedacht. Liegt aber ein optimales Programm vor, das noch schneller werden soll, dann hilft nur ein Basic-Compiler weiter. Dieser wandelt normale Basic-Programme in Maschinensprache um, bevor sie gestartet werden. Die Programme sind dann natürlich entsprechend schneller. Auf Seite 32 finden Sie übrigens einen Vergleich zwischen den gängigsten Compilern und dem Turbo-Basic-Compiler. Der hier abgedruckte Compiler hat bei der Gegenüberstellung die besten Zeiten erzielt. Wenn man dann noch in Betracht zieht, daß es den Turbo-Basic-Compiler fast zum Nulltarif gibt, dann ist er ein absolutes Muß.

# Ein Sprinter stellt sich vor: Compiler kontra Interpreter

Die heutigen Mikroprozessoren sind nicht in der Lage, ein Basic-Programm direkt auszuführen. Sie kennen nur Ihre interne Sprache, also die Maschinensprache. Aus diesem Grund ist der sogenannte Interpreter »zwischengeschaltet«. Er wandelt jede Basic-Programmzeile wieder in den Maschinencode um. So ist es einsichtig, daß Basic-Programme relativ langsam sind. Ein einfacher Basic-Befehl, wie beispielsweise »PRINT 3\*4« kann nämlich in der Übersetzung schon aus mehreren tausend Maschinenbefehlen bestehen.

Nun gibt es zwei Wege, ein Programm in einer Hochsprache, dazu zählt auch Basic, auszuführen. Entweder speichert man jeden Befehl, so wie er eingegeben wird. Nach dem Starten mit RUN führt ein entsprechendes Programm, der Interpreter also, für jeden einzelnen Befehl eine entsprechende Maschinensprache-Routine aus. Oder aber das gesamte Programm wird erst in Maschinensprache übersetzt, also compiliert, bevor es ausgeführt wird. Beide Verfahren haben Vor- und Nachteile.

Für den Interpreter spricht die einfache Bedienung. Es lassen sich mit ihm problemlos Programmänderungen durchführen, die dann auch schnell zu testen sind.

Mit dem Compiler zu arbeiten ist etwas umständlicher und die Fehlersuche aufwendiger. Hat man nämlich in seinem Basic-Programm einen nicht compilierbaren Befehl verwendet, dann muß man zwangsläufig den Interpreter wieder

laden oder den Computer ausschalten und erneut booten. Diesen Nachteil gleichen aber compilierte Programme gegenüber normalen Basic-Programmen durch ihre höhere Geschwindigkeit wieder aus.

Ein Interpreter benötigt weniger Speicherplatz als ein Compiler. Ebenso ist fast für jeden Compiler ein schneller Massenspeicher, also ein Diskettenlaufwerk, ein Muß. Es gibt sogar Compiler, die zwei Diskettenlaufwerke voraussetzen: eines für den Compiler und ein welteres für das zu compilierende Programm. Der Turbo-Basic-Compiler kommt allerdings mit nur einem Laufwerk aus.

Der Atari-Basic-Interpreter gehört einer besonders langsamen Spezies an. Als nämlich im Jahre 1979 der Interpreter entwickelt wurde, mußte er unbedingt in einem 8 KByte großen ROM Platz finden. Speicherplatz war zu der Zeit noch sehr teuer. Weiterhin rechnet das Atari-Basic mit BCD-Zahlen (Binary Coded Decimals), um Rundungsfehler zu vermeiden. Dies ist prinzipielt zeitaufwendiger als die normale binäre Arithmetik.

Turbo-Basic XL nimmt 18 KByte in Anspruch. Es enthält spezielle BCD-Routinen. Solche Algorithmen sind zwar recht umfangreich, dafür aber sehr schnell.

# Compiler zweigeteilt

Der Turbo-Basic-Compiler besteht aus zwei getrennten Programmen. Erstens aus dem elgentlichen Compiler, der die Übersetzung von Basic-Programmen übernimmt, und zweitens aus der Runtime-Bibliothek. Dieser Programmtell enthält die Arlthmetik- sowie andere Routinen, die zum Lauf des Compilats nötig sind. So ist sichergestellt, daß sich auch lange Programme compilieren lassen. Zum Zeitpunkt des Compilierens müssen sich also nicht der Compiler und das Runtime-Paket gleichzeitig im Speicher befinden. Weiterhin spart man sich so auch Diskettenplatz.

Der Compiler ist mit seinen etwa 20000 Byte sehr umfangreich. Um nun überhaupt damit arbeiten zu können, sind noch einige Dinge vorauszuschicken. Sie benötigen unbedingt ein Diskettenlaufwerk (mit Kassettenrecorder funktionieren weder Compiler noch Turbo-Basic XL). Außerdem müssen Sie, bevor Sie mit dem Abtippen der beiden Programme beginnen, unbedingt AMPEL (Atari-Maschinen-Programm-Eingabe-Listing) eingeben. Von Basic aus ist eine Eingabe des Compilers nämlich nicht möglich.

Mit AMPEL aber lassen sich reine Maschinenprogramme sehr schnell abtippen. Außerdem reduziert dieses Eingabeprogramm die Anzahl der einzugebenden Zeichen auf etwa 45 000. Entsprechende DATA-Listings oder Basic-Lader würden mehr als 80 000 Zeichen umfassen, und das ohne Fehlerkontrolle.

Wenn Sie also das Eingabeprogramm »AMPEL« vorliegen haben, starten Sie mit RUN und RETURN. Wählen Sie dann einen Namen, unter dem Sie das einzugebende Programm speichern möchten. Die Gerätebezeichnung ist bereits vor-



Bild 1. Das Titelbild des Turbo-Basic-Compilers

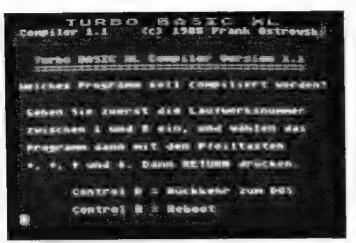


Bild 2. Nach dem Laden erscheint das Erklärungsmenü. Mit der Taste 1 können Sie sich das Inhaltsverzeichnis der Diskette in Laufwerk Nummer 1 betrachten.

PROG	RAMM-STECKBRIEF
Programmname	Turbo-Basic-Compiler
Programmtyp	Programmiersprache
Programmiersprache	Maschinensprache
Programmlänge	20809 Byte
für Computer	800 XL/130XE
zusätzliche Hardware	Diskettenlaufwerk
Eingabehilfe	AMPEL
Bemerkung	Der Compiler ist in zwei Telle auf- geteilt. Einerseits in den Compiler und andererseits in das Runtime-Paket
Leserservice	Diskette (COMPILER.COM und RUNTIME.COM)

gegeben, nämlich »D1:« für Diskettenlaufwerk Nummer eins. Anschließend werden Sie nach der Anzahl der Bytes gefragt. Entnehmen Sie diese Zahl vom Ende des jeweiligen AMPEL-Listings. Sollten Sie bereits einen Teil des Programms vorliegen haben, geben Sie bitte unbedingt den zuvor gewählten Namen ein. Dann wird der bereits eingegebene Teil, den Sie mit CONTROL-S gespeichert haben, wieder in den RAM-Speicher geladen, und Sie können an der Stelle fortfahren, an der Sie aufgehört hatten.

Bedenken Sie bitte, daß Sie bei der Benennung des Programms unterschiedliche Namen verwenden müssen. Wählen Sie also für Listing 1 beispielsweise den Namen »COMPILER.OBJ« und für das zweite »RUNTIME.OBJ«. Somit sind eventuelle Verwechslungen weitestgehend ausgeschlossen.

Wenn beide Programme komplett eingegeben sind, sollten Sie sich unbedingt eine oder besser noch mehrere Sicherheltskopien anfertigen. Auch ist zu empfehlen, während der Eingabe der Programme öfter CONTROL-S zu betätigen, um das Programm im Speicher auf Diskette zu speichern. Das dauert nur wenige Sekunden, für das erneute Eingeben der Programme nach einem Stromausfall benötigen Sie jedoch ein Vielfaches der Zeit.

Verwenden Sie auch für den eigentlichen Compiler und das Runtime-Programm jeweils eine eigene Diskette. Beide Programme sind nämlich als »AUTORUN.SYS«-Files ausgelegt. Das bedeutet, daß sich beide Programme nach dem Einschalten des Computers automatisch laden und starten können. Für ein Autostart-Programm ist Voraussetzung, daß es die Bezeichnung »AUTORUN.SYS« auf Diskette hat. Dann muß sich nur noch das File DOS.SYS (entweder DOS 2.0 oder DOS 2.5) und gegebenenfalls DUP.SYS auf der Diskette



Bild 3. Das Inhaltsverzeichnis präsentiert sich in dieser Form. Mit den Pfeiltasten wählen Sie das zu compilierende Programm aus.

befinden, und schon laden und starten sich die Programme nach dem Einschalten des Computers automatisch.

Das Umbenennen eines Dateinamens ist übrigens sehr einfach. Nehmen wir an, daß sich der Compiler unter dem Namen »COMPILER OBJ« auf Diskette befindet. Das Programm soll nun in »AUTORUN.SYS« umbenannt werden. Legen Sie also eine Diskette mit den Files DOS.SYS und DUP.SYS ins Laufwerk ein, und schaften Sie die Stromversorgung des Computers an. Nach der READY-Meldung geben Sie DOS ein, um in das entsprechende Menü zu gelangen. Wählen Sie jetzt die Funktion E für »Rename File«, und geben Sie den alten und, mit einem Komma getrennt, den neuen Programmnamen ein. Der neue Programmname lautet also In unserem Beispiel »AUTORUN.SYS«. Jetzt haben Sie ein Autostart-Programm vorliegen. Dies funktioniert jedoch nur mit speziell für einen Autostart ausgelegten Maschinen-Programmen.

Für Ihre Mühe, das Programm eingetippt zu haben, werden Sie – falls Sie auch an genügend Sicherheitskopien gedacht haben – belohnt. Sie verfügen nämlich damit über einen Compiler, der sowohl beim Compilieren als auch bei der Programmausführung besonders schnell ist.

# **Bedienung des Compilers**

Der Umgang mit dem Turbo-Basic-Compiler ist sehr einfach. Es ist nicht nötig, laufend mit der Bedienungsanleitung in der Hand zu arbeiten. Allerdings sollten Sie sich diese Beschreibung zumindest einmal aufmerksam durchlesen. Übrigens, haben Sie schon Ihren Compiler und das Runtime-Paket auf jeweils einer Diskette vorliegen?

Wenn Sie also ein Programm compilieren möchten, sollte es mit SAVE gespeichert sein. Mit LIST gespeicherte Programme lassen sich mit dem Turbo-Basic-Compiler nicht compilieren.

Schieben Sie also einfach die Compiler-Diskette in Laufwerk Nummer D1:, und schalten Sie anschließend den Computer ein. Dann wird automatisch das DOS und das AUTO-RUN SYS-File mit dem Compiler geladen. Zunächst sollte das Titelbild (Bild 1) erscheinen. Dieser Compiler meldet sich mit einer kurzen Bedienungsanlertung (Bild 2). Jetzt können Sie die Compiler-Diskette aus dem Laufwerk nehmen, sie wird für den folgenden Vorgang nicht mehr benötigt. Legen Sie dann die Diskette mit dem zu compilierenden Programm ein, und drücken Sie die Taste 1. Sollte Ihnen mehr als ein Laufwerk zur Verfügung stehen, geben Sie die entsprechende Laufwerknummer an. Anschließend wird der Bildschirm gelöscht. Es verbleiben nur drei Zeilen am oberen

Bildschirmrand. Auf dem Bildschirm erscheint daraufhin das Inhaltsverzeichnis der Diskette. Aus Platzmangel mußte auf die Angabe der belegten Sektoren der einzelnen Files verzichtet werden. Sollte die Diskette leer sein, gelangen Sie wieder zum Titelbild. Wenn sich das zu compilierende Programm nicht auf Diskette befindet, wechseln Sie sie einfach aus und betätigen dann noch einmal die Taste 1.

Der erste Programmname, in der oberen linken Bildschirmecke, wird invers dargestellt. Bild 3 verdeutlicht dies. Die Ausgabe erfolgt dabei mit dunklen Buchstaben auf hellem Hintergrund, so als wäre der Cursor viel breiter. Diesen »Cursor« können Sie mit den Pfeiltasten, ohne CONTROL-Taste, bewegen. Wenn der richtige Filename invertiert ist, drücken Sie RETURN, und schon beginnt die Compilierung. Während dieses Vorgangs wird nur von Diskette gelesen, aber keine Zwischenfiles auf Diskette abgelegt.

Beim Turbo-Basic-Compiler handelt es sich um einen »Two Pass Compiler«. Das heißt, die Compilierung erfolgt in zwei Durchgängen. Dabei muß das Programm nur einmal von Diskette gelesen werden. War das Programm problemlos zu compilieren, erfolgen die Meldung »Keine Fehler« und eine Angabe über die Programmlänge. Die Frage nach dem Programmnamen, unter dem das compilierte Programm auf Diskette gespeichert werden soll, schließt das Ganze ab. Dabei sind die Gerätebezeichnung, also D1 für Laufwerk Nummer eins, und der Fileextender (der Text, der nach dem Punkt folgt), also ».FBC« schon vorgegeben. Wenn Sie mehr als ein Laufwerk besitzen, können Sie an dieser Stelle die entsprechende Taste für die Laufwerknummer drücken.

Geben Sie jetzt den Filenamen ein. Das erste Zeichen muß dabei ein Buchstabe sein, und die Gesamtlänge darf acht Zeichen nicht überschreiten. Bei einer fehlerhaften Eingabe können Sie mit der DELETE/BACKSPACE-Taste korrigieren. Nach Betätigung der RETURN-Taste wird das endgültige Compilat unter dem eingegebenen Namen auf der Diskette gespeichert, worauf sich auch die Runtime-Bibliothek (als AUTORUN.SYS-File) befinden sollte. Wenn Sie Ihrem compilierten Programm den Namen AUTORUN.FBC geben, wird es beim Booten automatisch geladen und gestartet.

Ist das Speichern beendet, werden Sie gefragt, ob Sie das Programm noch einmal speichern möchten. Drücken Sie »J« für ja, sonst »N« für nein. Mit »N« gelangen Sie wieder ins Titelbild, das nach dem Booten des Compilers auf dem Bildschirm zu sehen war. Jetzt können Sie wieder die Taste 1 drücken, falls Sie ein weiteres Programm compilieren möchten. Mit CONTROL-D gelangt man ins DOS, und CONTROL-R entspricht dem Aus- und Einschalten des Computers.

Zum Ausführen des compilierten Programms müssen Sie die Diskette mit der Runtime-Bibliothek einlegen und booten. Dazu können Sie die CONTROL-R-Funktion im Compiler wählen oder den Computer aus- und wieder einschalten. Wenn sich auf der Diskette ein File mit dem Namen AUTO-RUN.FBC befindet, wird dieses geladen und gestartet. Andernfalls erscheint die Meldung »Fehler 170 in 12345 (\$AAAA)«. Die Fehlernummer 170 steht für »File not found«. Es wurde also versucht, ein nicht auf Diskette befindliches Programm zu laden. Nähere Informationen hierzu kann man auch der Basic- oder DOS-Anleitung entnehmen. Sollte also diese Fehlermeldung erscheinen, ist die Zeilennummer zufällig und der in Klammern angegebene 6502-Programmzählerstand unwichtig. Nach jeder Fehlermeldung, dem Programmende oder END, STOP und DOS-Befehlen in compilierten Programm erscheint »Programmende: Dos, Run oder Load?«. Wenn Sie jetzt die Taste D drücken, wird das DOS aufgerufen. R wirkt wie RUN vom Basic-Interpreter aus, und Llädt ein Programm von Diskette (wie »RUN "D:Filename.ext" « unter Basic). Alle anderen Tasten werden ignoriert. Nach L erscheint » "Filename D: " « auf dem Bildschirm. Sie können jetzt den Namen eines compilierten Programms eingeben.

Der Extender ».FBC« wird automatisch angefügt, allerdings nicht angezeigt (sonst wäre das Runtime-File zu lang). Wie unter DOS können Sie auch ein Prefix wie D2: angeben.

### Nicht compilierbare Befehle

Es gibt eine Reihe von Befehlen, die der Turbo-Basic-Compiler nicht unterstützt. Diese sind im einzelnen: LIST, ENTER, \*L, DEL, RENUM, DUMP, TRACE, CONT, LOAD, SAVE, CLOAD, CSAVE, NEW und ERROR. Sämtliche Variablennamen sowie REM-Zeilen werden beim Compilieren nicht berücksichtigt.

Andererseits compiliert der Turbo-Basic-Compiler eine Reihe von Befehlen, die andere, wie beispielsweise der ABCoder MMG-Compiler, nicht unterstützen. Dabei handelt es sich besonders um die Turbo-Basic XL-spezifischen Befehle, berechnete GOTO-Anweisungen, GOSUB, TRAP und RESTORE sowie DIM. Berechnete Sprungziele nach ON werden allerdings nicht verarbeitet.

Außerdem stellt der Turbo-Basic-Compiler noch gewisse Ansprüche an die Strukturierung von Programmen. So dürfen in einem Programm elner FOR-NEXT-Schleife auf keinen Fall mehrere NEXT-Anweisungen zugeordnet sein. Das gleiche gilt für die REPEAT-UNTIŁ-, WHILE-WEND- und DO-LOOP-Schleifen. Sie dürfen weiterhin nicht durch GOTO-Anweisungen zerstückelt werden. Fehlermeldungen wie >110 FOR fehlt«, >120 NEXT fehlt«, >800 UNTIL fehlt« sind auf solche ungünstigen Programm-Konstruktionen zurückzuführen.

Hier zwei Beispiele für einen sehr ungeschickten Programmierstil:

```
100 GOTO 120

110 NEXT I:GOTO 130

120 FOR I=1 TO 5:PRINT I:GOTO 110

130 REM

100 FOR I=1 TO 10

110 IF A THEN ? -I:NEXT I:GOTO 130

120 ? I:NEXT I

130 REM
```

Diese beiden Programme werden vom Interpreter anstandslos ausgeführt. Beim Compilieren ist aber die Reihenfolge der FOR- und NEXT-Befehle im Programmtext ausschlaggebend und nicht die Reihenfolge, die sich bei Verwendung von GOTO-Anweisungen ergibt. Der Turbo-Basic XL-Befehl GO # (GOTO Marke) ist in den oben gezeigten Beispielen ebenfalls nicht zulässig.

Das erste Programm ist ein typisches Beispiel für den sogenannten »Spaghetticode«. Der Programmfluß läßt sich ähnlich gut verfolgen wie die Schlingen einer Nudel auf einem Teller. Das zweite Beispiel ist schon besser programmiert. Nutzt man aber die Fähigkeiten von Turbo-Basic XL, nämlich strukturiert zu programmieren, voll aus, sähe das zweite Beispiel so aus:

```
100 FOR I=1 TO 10:IF A:? -I:ELSE:? I:ENDIF:NEXT I oder, noch übersichtlicher, so:
```

```
IF A
110
        ? -I
120
130
      ELSE
140
        ? I
150
      ENDIF
160 NEXT I
In Atari-Basic könnte das dann so aussehen:
100 FOR I=1 TO 10
110 IF A THEN ? -I:GOTO 125
120 ? I
125 NEXT I
```

100 FOR I=1 TO 10

130 REM

Auch das Atari-Basic-Beispiel wird vom Compiler anstandslos verarbeitet und erzeugt sogar genau den gleichen Code wie die strukturierte Turbo-Basic XL-Version. Die Übersichtlichkeit leidet aber hier schon durch den GOTO-Befehl in Zelle 110. Sie sollten wirklich nur dann so programmieren, wenn Sie auf diese Weise Ihr Programm besser überblicken können. Möchte man aber sein Programm durchgehend strukturiert gestalten, dürfen keine GOTO- sowie GOSUB-Befehle verwendet werden. Wer bereits Erfahrungen mit der Programmiersprache Pascal gesammelt hat, weiß, daß es hier solche Befehle überhaupt nicht gibt. In Turbo-Basic XL lassen sich die GOTO- und GOSUB-Befehle fast immer durch IF-ELSE-ENDIF, WHILE-WEND, REPEAT-UNTIL oder DO-EXIT-LOOP ersetzen.

Weitere Fehler treten auf, wenn bei POKE-, DPOKE-, PEEK- oder auch PLOT- oder GRAPHICS-Befehlen zu große, oder zu kleine Zahlen verwendet werden; also bei allen Befehlen, die Integer-Parameter verwenden. Zum Beispiel: »POKE 100,300«, »DPOKE 70000.1«, »PLOT 100,300«, »? PEEK (-1)« oder »GRAPHICS -5« sind also nicht zulässig. Der Turbo-Basic-Compiler wandelt nämlich bei solchen Befehlen konstante Zahlen selbständig in die kompaktere Integer-Darstellung um. Dies erledigt der Basic-Interpreter erst zur Laufzeit und gibt dann die Fehlermeldung »ERROR 3« aus. Diese Umwandlung kostet den Interpreter viel Zeit und ist einer der Gründe dafür, warum compilierte Programme schneller sind. Als Fehlermeldungen gibt es noch »Integer <0 oder >65535« und »Integer > 255«.

Für Atari-Computer werden neben dem Standard-Atari-Basic und Turbo-Basic XL noch einige weitere Basic-Interpreter angeboten. Dazu zählen Basic A+, Basic XL und Basic XE. Diese Interpreter verwenden andere Token als TurboBasic XL. Sollten Sie also ein Programm, das mit einem dieser Interpreter geschrieben wurde, compilieren, verwechselt der Turbo-Basic-Compiler zwangsläufig die Token, und somit die Befehle. Als Resultat erhalten Sie dann ein vollkommen falsches Compilat. In den meisten Fällen bekommen Sie dann die Meldung »Compiler Fehler« oder »Token #xxx \*\*\* Muell\*«.

Eine weitere Fehlermeldung ist »Ausdruck zu komplex«. Sie tritt auf, wenn bei der Übersetzung einer Berechnung ein spezieller, interner Umwandlungspuffer des Compllers überläuft. Dieser Fehler ist aber – auch nach intensivem Testen des Compllers – noch nicht aufgetreten.

(Frank Ostrowski/wb)

# Vorsicht, Fehler QJSA und QJSB!

Bei der Entwicklung des Turbo-Basic-Compilers wurden die Fehlermeldungen »QJSA« und »QJSB« eingebaut. Diese dürfen aber unter keinen Umständen auftreten. Sollten Sie solche Fehler erhalten, teilen Sie dles bitte umgehend der Happy-Computer-Redaktion mit. Legen Sie dann Ihr Programm auf Diskette bei. Wenn Sie den Turbo-Basic-Compiler abgetippt haben, kann auch ein unentdeckter Tippfehler die Ursache seln. Wenden Sie sich aber auf jeden Fall an:

Redaktion Happy-Computer z. Hd. Werner Breuer »Compiler-Fehler« Hans-Pinsel-Str. 10a 8013 Haar

```
0000:FF FF 00 3E AF 3E 00 18<1A>
                                           0178:28 11 20 20 20 11 20 20(F9)
                                                                                     02F0:02 02 02 02 02 02 02 02 02<F4>
                            22<53>
0008:18 FF
            3C
               3C 66
                      42
                         00
                                           0160:11 98
                                                       20 11 20 20
                                                                    20
                                                                       20<E0>
                                                                                     02F8:02 02 02
                                                                                                    82 82 82
                                                                                                              02
                                                                                                                 B2<FC>
                  76
                            00<7B>
0010:00
                                                       20
                                                                                              02
                                                                                                 02
                                                                                                    82
                                                                                                        41
                                                                                                           38
                                                                                                                  88<8D>
            66 7E 66
3E 30 30
9918:18
        30
                      66
                         88
                            3E < D6 >
                                           0190:11 20
                                                      11
                                                          11 11
                                                                 14
                                                                    20
                                                                       11<80>
                                                                                     0208:00 00
                                                                                                 34 35 32
                                                                                                           22
        33
                            70<30>
                                           0198:20 20
                                                         20 20
                                                                28
                                                                                              21
0020:73
                      78
                         00
                                                      11
                                                                    11
                                                                       11(87)
                                                                                     R310:22
                                                                                                 33
                                                                                                    29 23
                                                                                                           an
                                                                                                              38
                                                                                                                 20<10>
0028:37
        33
               30
                  30
                      78
                                                                                     0318100 00
            3E
                         88
                            00(BE)
                                           01A0120 20
                                                      11
                                                          11
                                                             11
                                                                14
                                                                                                 90
                                                                                                    23 6F
                                                                       20(10)
                                                                                                           6D
                                                                                                              79
                                                                                                                  69(B5)
               3C
                  39
                                           81A8:11 20
                                                          20 20
                                                                       28<D6>
                                                                                              65
            36
                                                                                     0320: 6C
                                                                                                 72
                                                                                                           ØE
                                                                                                              11
                                                                                                       11
                                                                                                                  99<C6>
                                                         20 20
20 11
0038:01 0F
            (B33)
               03 F6
                      E4
                         22
                            自食くじこ〉
                                           01B0:11 28
                                                      11
                                                                11
                                                                    20
                                                                       11(85)
                                                                                     0328:00 00
                                                                                                 22
                                                                                                    08 43
                                                                                                           09
                                                                                                                  11(F6)
                                           2188:20 20
0040:00 FO CO CO 60
                      20
                            B1(B4)
                                                      28
                                                                                                           72
72
                         00
                                                                20
                                                                    20
                                                                       11(31)
                                                                                     D330:19 18
                                                                                                 15
                                                                                                    69 26
                                                                                                              61
                                                                                                                  6E<97>
                      01
                                                       20
                                                             20
                                                                       11<BA>
0048:03 96
            06
               96 93
                            E1(69)
                                           01C0:20 20
                                                          11
                                                                20
                                                                                     8338:68 88
                                                                                                       74
                                                                                                    73
                                                                                                                  77<C4>
                                                                                                 2F
                                                                                                              6F
0050:33 06
            86
               06
                  33
                                                       20
                                                          7B
                                                             20
                                                                15
                                                                       11(48)
                                                                                     0340:73
                      E1
                            C7<70>
                                           01C8:15 13
                                                                                              68
                                                                                                       60
                                                                                                           OD
                                                                                                              61
                                          0100:14 20
0108:20 20
                                                                    20
20
0058:63 33
            36 36 66
                      CE
                         00
                            @E<EB>
                                                      15
                                                         11 11
                                                                14
                                                                       11<02>
                                                                                     0348:24 E4
                                                                                                 AC 25
                                                                                                       E4
                                                                                                           EB.
                                                                                                              DØ
                                                                                                                 @1<74>
        9C F6
                                                                                                 24
                            7E<89>
                                                          28
                                                             11 20
                                                                       20<7C>
                                                                                                    21
0060:9C
               66 66
                      67
                         20
                                                      11
                                                                                     0350:08 8E
                                                                                                       80
                                                                                                           25
                                                                                                              21
                                                                                                                 A9(FB)
               20
                  30
                                                                                                 BE D4
                                                                                                           A9
        33
            3E
                      78
                         98
                            67(DF>
                                           01E0:12 11
                                                          20
                                                                11
                                                                       20(A3)
                                                                                                       78
                                                                                                              FE
0068:33
                                                       13
                                                             11
                                                                                     0356:00
                                                                                             BD
                                                                                                                  9D< 73>
                            FF(EA)
                                                                                                 A9 CB
                                                                                                       BD
0070166
            66
                                           Ø1E8:11 11
                                                       13
                                                                       13<62>
                                                                                     0360:01
                                                                                             D3
                                                                                                                  A9<60>
                                                      7F
6E
                                                          7F
0070:31
        31
            31 31
                  31
                      7B
                         80
                            FREAD
                                           01F0:98 98
                                                             56
                                                                65
                                                                       73(A5)
                                                                                     03A8: 20
                                                                                             BD
                                                                                                 FB
                                                                                                    FF
                                                                                                        A9
                                                                                                           FØ
                                                                                                              GB
                                                                                                                 FE<15>
                                                             31
                                                                ZE.
                                                                    31
                                           01F8:69 6F
                                                          28
                                                                       9B(DA)
0080:91 B1
            F1
               81
                  81 FB
                            FB<CF>
                                                                                                 20 BD FF
                         60
                                                                                     0370 rFF
                                                                                              A9
                                                                                                           FF
                                                                                                              49
                                                                                                                 CC< GE>
                                                   7F
9088:8C
        80
           FB B0
                  98
                      CC
                         55
                            AAK3E>
                                           0200:9B
                                                       20
                                                         28
                                                                29
                                                                       31<68>
                                                                                             65
                                                                                                    A9
                                                                                                       EØ
                                                                                                           20
                                                                                     Ø378:20
                                                                                                 60
                                                                                                              65
                                                                                                                 60<B6>
            55
0090155
        AA
               AA
                  55
                      AA.
                         81
                            02<50>
                                           0220:39
                                                   38
                                                      35
                                                         28 46
                                                                72
                                                                    61
                                                                       AE(82)
                                                                                     0380:A9 40
                                                                                                 αB
                                                                                                                  FF<4E>
                                                          73 74
                                                                72 6F
E2 02
                                                                                     0388:80 01 D3 20 88 60
0390:85 09 A5 0C A4 0D
MASHIRS MA
            15
               20
                  55 AA
                         000
                            BB(94)
                                           8218:68 28
                                                      4F
                                                                       77(A2)
                                                                                                              A9
                                                                                                                  四四<11>
20A0: 42 A0
            50 AB
                  54
                      AA
                         55
                            AA<17>
                                                          98
                                                             9B
                                                      69
                                           @218:73
                                                   άB
                                                                       E3(CE)
                                                                                                              8D
                                                                                                                 7E<E3>
00A8:54
        AB
            50 A0
                  40
                                                                                                 7F
                                           0220:02
                                                             28
                                                                98
                                                                                     0398:20 SC
                                                                                                    20 A2
                                                                                                           01
                                                                                                              8E
                                                                       2億(84)
                                                                                                                 F8<89>
                            30<B6>
0090:15 0A 05
               02 01 00
                         20
                                           0228:40
                                                   15
                                                      A9
                                                         7D A0
                                                                20 85
                                                                       0C(11>
                                                                                     83A8183 86 89 86
                                                                                                       98
                                                                                                                  44(CB)
80BB162
        31 A2 00 BD 00
                         EØ
                            9B<AD>
                                           0230:84 ND A9
                                                          00 65 52 85
                                                                       A2<E5>
                                                                                     03A8:02 4C 80 20 65
                                                                                                           05
                                                                                                              AØ
                                                                                                                 20<4D>
99C9:99
        3C
            BD
                                                                                     0380:84 D4 A2 04 A9
               99
                  E1
                      9D
                         99
                            3D<1D>
                                           0238:A9 FE
                                                      BD
                                                         01 D3 4C
                                                                    00
                                                                       CØ(39)
                                                                                                           FF
                                                                                                              BD.
                                                                                                                 @1<C5>
                      3F
        6969
            E3
                                           0240:EE 01
                                                      D3
                                                         6C
                                                                       01<090
                                                                                     @3B8: D3 B1
                                                                                                 D4
                                                                                                       88
                                                                                                           05
                                                                                                              CE
                                                                                                                 DØ<CD>
OODO: EB A9
            10 8D C6
                     02
                         49
                            SPZAES
                                           024B: D3 4C
                                                       77
                                                          E4
                                                             44
                                                                31 3A
                                                                       2A(37)
                                                                                     03C01FB CE
                                                                                                 01 D3 B9
                                                                                                           99
                                                                                                              25
                                                                                                                  91<24>
0008:80 C5 02 80 F0 02
                         BD
                            F4<FE>
                                           0250: 2E 2A
                                                      9B
                                                         50
                                                             3A 98 45
                                                                       3A<4B>
                                                                                     03C0:D4 C8 D0 F8 E6 D5
03D0:E3 60 A2 10 A9 D6
                                                                                                              CA
                                                                                                                 DB< RA>
00E0:02 A9
            47
               8D 44
                     23
                                                      Ø1 03
                         A9
                            30<7D>
                                           0258:98 EE
                                                             20 56
                                                                    E4
                                                                       08<98>
                                                                                                              9D
                                                                                                                 44<D1>
00E818D
        45
                  10
                      BD
                                           0260:48 AD
                                                      2)1
                                                          D3
                                                             29 FC
                                                                       @2<@C>
                                                                                     03D8:03 A9
                                                                                                 62
                                                                                                    9D
                                                                                                       45
                                                                                                           Ø3
                                                                                                                  49<0E>
88F81A9 81 BD 49 83 A9
                         ZB.
                            BD<29>
                                           0268:8D 01
                                                      D3
                                                         68
                                                             28 60
                                                                    68
                                                                                     03E0:03 A9 04 90 48
                                                                                                                 @7<93>
                                                                       AA<01>
                                                                                                           83
               56
                     BE
MMF8:42 M3
            20
                  E4
                         FØ
                            02(EA)
                                          0270xCF 01 D3
                                                         68 40 2C
                                                                    ØF.
                                                                       D4<215
                                                                                     83E8: 9D 42 83 28
                                                                                                       56
                                                                                                           E4
                                                                                                              300
                                                                                                                 5A<E4>
            9B
0100:60
        2D
               98
                  7F
                      7F
                         20
                            22< BC>
                                           0278:10 03 6C
                                                         00 02 48
                                                                    BA
                                                                                     03F0: A9
                                                                                             Ø6
                                                                                                 85 DA
                                                                                                       9D
                                                                                                           44
                                                                                                                 A9(C2)
                                                                       48<DE>
                                                                                                              B3
9198:01
        02 03 04 05 06
                            9B<94>
                                          0280: A7
                                                   20
                                                      48
                                                         A9
                                                             C4
                                                                48
                                                                       BD< 62>
                                                                                     03F8:61 85 DB 9D
                                                                                                       45
                                                                                                           63
                                                                                                                  AS(BF)
0110:7F
        7F
            08 09
                  MA
                     ØB.
                         ØC.
                            MD(E4)
                                          0288:05 01 48
                                                         DB 48 9A
                                                                    48
                                                                       98<62>
                                                                                     0400:DB E5 D6
                                                                                                    85 DC
                                                                                                           A5
                                                                                                              09
                                                                                                                 E5(C2)
011810E 0F
            10
               9B
                  9B
                     20
                         20
                            20<4C>
                                          8298:48 EE 81
                                                         D3 SD RF
                                                                    D4
                                                                       ACKSED
                                                                                     0408:D7
                                                                                             85 DD E6 DC
                                                                                                           DB
                                                                                                              02
                                                                                                                 E6<1F>
0170:20
        20
            20 20
                  20
                      20
                            20<11>
                                          0298:22 02
                                                      48
                                                         A9
                                                             20 48 A9
                                                                                     0410:DD A5
0418:9D 49
                                                                                                    9D
                                                                                                 DC
                                                                                                       48
                                                                                                           23
                                                                       CA<FB>
                                                                                                              A5
                                                                                                                 DD<A9>
            72
                      20
                         42
                            41<CF>
                                                         01
                                                             D3 6C
                                                                                     0418:9D
                                                                                                 03 20
                                                                                                       56
                                                                                                           E4
                                                                                                              30
@128:54
                                          02A0:48 08
                                                      EE
                                                                                                                 32<36>
                                                                    CE
0130:53 49
            43 28 58 4C
                         98
                            9B<5A>
                                          02A8:EE 01 D3 20
                                                             28 21
                                                                       01<B1>
                                                                                     0420:CE 01 D3 A0 00 A6
                                                                                                              DD F0<E5>
0138:20 12 11 11 13 20
                            11<50>
                         12
                                                                                                                 F9(2E)
                                          0280:D3 60
                                                      BD
                                                         47
                                                             03 4B BD
                                                                       46<2A>
                                                                                     @428:0E B1 DA 91 D6 C8
                                                                                                              DØ
0140:11 13
            20
                  13
                      12
                            20(C7)
                                          0200:03
                                                   48
                                                         4A 23 85
                                                                                                 E6 DB
              11
                         11
                                                      BD
                                                                       BD<88>
                                                                                     0430:E6 D7
                                                                    2A
                                                                                                       CA DØ
                                                                                                                 A6<82>
                                                                                                              FZ
0148:11
                            20<9F>
                                                   B3
                                                         ZB
                                                                                                                 CB<BE>
            11 13 15
                      11
                         14
                                          02C0:48
                                                      85
                                                             98 AØ
                                                                       60<E0>
                                                                                     6438:DC FØ
                                                                                                 80
                                                                                                    B1 DA
                                                                                                           91
                                                                                                              06
0150:11 20
           28 28 11
                      11
                         14
                            28(FD)
                                          02CB: EE
                                                   281
                                                      D3
                                                         06
                                                             2A
                                                                20
                                                                                     0440: CA
                                                                                             DØ
                                                                                                 FB
                                                                                                    A9
                                                                    EZ
                                                                       F6<62>
                                                                                                       FF
                                                                                                                  D3<B9>
                  98
            11 13
                     20
                            20(19)
                                                                                                           42
E2
0150:11
        11
                         11
                                          02001 CE
                                                   PL 1
                                                      D3
                                                         60
                                                             EE
                                                                Ø1 D3
                                                                       28(95)
                                                                                     2448:3N
                                                                                             RA A9
                                                                                                    ØC
                                                                                                       90
                                                                                                              93
                                                                                                                 4C<7B>
0160:20
        29
            20 11
                  20
                         11
                                          02DB:02
                                                   D9
                                                         @1
                                                             D3
                                                                A5 D4
                                                                       A4<7E>
                                                                                     0450:56
                                                                                              E4
                                                                                                    ØA.
                                                      CE
                                                                                                 6C
                                                                                                       99
                                                                                                              02
                                                                                                                 E3(BB)
        15
            14 11
                  20
                      11
                         20
                            20<71>
                                                                                                    20
Ø168:11
                                                             40
                                                                                              00
                                                                                                        CØ
                                                                                                           3D
                                                                                                              CB
                                                                                                                  20(13)
0170:11 20 11 20 20 11 20 20 (88)
                                          02E8:00 02 00 02 00 42 B8 BC<C2>
                                                                                     0460:CA CB 18 A5 58 69 78 8D<9F>
```

Listing 1. Turbo-Basic-Compiler (Länge: 9933 Byte). Geben Sie das Programm bitte mit AMPEL von Seite 87 ein.



0468:44 21 AS 59 69 20 BD 45<1D>	9748: DB B0 C9 68 A2 80 A9 6F(E0)	0077 P7 O. T. C. T. C.
0470:21 A9 38 BD 30 02 A9 21 <c0></c0>	0750:95 00 A9 22 95 01 EB E8<60>	0A28:D7 C6 98 C6 98 C8 99 80 <f7></f7>
0478:80 31 02 A9 03 85 54 20<61>		0A30:DA A5 9E CE 99 00 DA 84<8F>
0480:10 01 AZ FF 9A 20 BE F1(CA)	0758:E0 8C 98 F2 A9 FF A0 00<13>	BA38:90 40 58 05 28 48 01 89(38)
MANDE OF THE STATE	0760:84 90 84 91 84 BC 84 8D<7F>	@A46:00 30 ZE F0 29 C9 @F 98<3F>
0488:20 ED C2 20 54 C2 20 5C<57>	2768:85 92 85 93 99 88 DC 99 <ec></ec>	MINITURE FR BD 68 AB 88 28 CF(83)
0490:C1 4C 2F C0 20 EF F8 20< 0A>	0770:00 DD 99 00 DB CB D0 F4(5A)	8A58:C7 28 26 C6 A8 86 D8 BE<9D>
9498:F1 F4 29 FA E4 20 FA E4<14>	0778:20 D5 EC A9 FF 91 AB 60(C5)	MANUTURA B1 20 CC C7 20 26 C6(D4)
04A0:A5 90 05 91 F0 0A A5 90(8B)	0780:07 C5 AD EE DD EF 4A EA(F9)	50 AU CO
04A8:A4 91 20 E8 C1 4C 5A C8(19)	0788:28 E4 28 E4 69 EA 12 EE(D1)	WANTEN 48 C1 A8 F0 06 20 48(C8)
04B0:20 EF E4 4B 65 69 6E 65(CB)	8798:F4 EE 32 EF 1A EA 1A EA(DF)	@A681C1 88 DØ FA 18 68 20 48<63>
0488:00 20 EF E4 20 46 65 68(31)	MANUAL TO TO TO TO THE EN THE ENCOR!	0A78:C1 49 80 48 A0 82 20 CF<42>
04C0:6C 65 72 98 50 72 6F 67<86>	0798:1A EA 52 FB 7C E9 28 E4<31>	0A78: C7 68 AB BE 00 DB 86 98(48)
GACE 72 (4 (7 15 (6 (7 6)	07A0:5C EA 4E EC 7F E9 67 E9(1D)	ØA80:20 F3 E9 18 60 18 A9 80<44>
04CB: 72 61 6D 6D 6C 61 65 6E<6E>	07A8:5C EA 82 E9 28 E4 FE FB(F3)	0A88: 45 97 08 20 F3 E9 28 A9(74)
0400:67 65 20 30 20 00 38 A5<53>	0780:28 E4 28 E4 94 EF BC EF(87)	8A98:84 69 88 4C F3 E9 C4 C7<8F>
0408:88 E9 6F AA A5 89 E9 22(86)	07BB:CF EF F3 F6 17 F1 63 EB(73)	9A98:C4 C7 C4 C7 C4 C7 C4 C7(CD)
04E0:A8 8A 20 E8 C1 20 EF E4 <cf></cf>	07C0:7C EA 64 E9 ED EF 48 FB(C5)	
04E8:20 42 79 74 65 73 98 00(88)	87C8:79 E9 68 EC 82 E9 73 E9(FF)	MAARIC4 C7 BD C7 BD C7 BD C7 AD>
04F0:A5 90 05 91 D0 06 20 47(4D)	9708:7C EA BA EF 63 EF 43 EC(B8)	BAAB: BD C7 BD C7 B1 C8 C4 C7<4A>
04F8:F7 4C 23 C0 A2 18 20 83(4F)	GADOLEC CHICH EL 02 EL 92 EC(BB)	0A90:C4 C7 22 C8 C9 C8 00 20<86>
0500:C4 A7 00 85 A2 20 EF E4<3A>	07D0:F6 EB F9 EB 82 E9 F3 EB(F3)	MANUAL DE DE C4 C7 C4 C7 C4 C7 C41>
	07E0: 2E F0 B2 EF 12 F0 74 EA(E9)	0AC0:C4 C7 C4 C7 C4 C7 22 C8(A2)
0508:98 98 42 69 74 74 65 20(AD)	07E8:28 E4 28 E4 69 EA 28 E4<05>	BACB: 9B C7 49 CB 4C CB 46 CB(37)
0510:65 69 6E 65 20 54 61 73 <d1></d1>	27F8:99 EB BD EC CO EC B1 EA(16)	@ADM: C9 C8 46 C8 C7 C8 95 C7(EE)
0518:74 65 20 64 72 8A 63 68(E3)	97F8: DC EC 68 ED 89 ED 84 EE(8D)	@ADB: F5 C7 @B CB A6 C7 98 C7<3F>
0520:45 AE 90 20 F7 C0 4C 23(EC)	8888:88 EE 62 EE AD EC AA EC(51)	BAED: A6 C7 A6 C7 9B C7 9B C7 (CF)
0528:C0 AZ 00 20 83 C4 A9 C0(F5)	8808:F0 E8 DF EC 66 ED D7 ED(E0)	CACOLAL CT OD CT OD CT TO CT TO
0530:05 6A A9 03 9D 42 03 A9(39)	8818: C5 F8 D4 F8 D7 F8 DA F8 5A>	BAE8: A1 C7 9B C7 9B C7 A6 C7<11>
0538:0C 9D 4A 03 A9 00 9D 48 <fc></fc>	BOIG-OD FO TA CO TO TO ON FECOM	@AF@:9B C7 9B C7 9B C7(53)
0540:03 AP AC 9D 44 03 AP 20(14)	8919:00 FØ 7A FØ EØ FØ 8A FØ(8A)	MAFB: 9B C7 9B C7 9B C7 BB C7(85)
0540 DD 45 03 DD 45 00 TO TO	8828:98 F8 76 E9 52 EA 28 E4<24>	0800:80 C7 80 C7 80 C7 A1 C7<47>
0548:90 45 03 20 AF 20 30 D9(83)	8828:07 C5 28 E4 28 E4 28 E4<83>	0808:A1 C7 A1 C7 23 CB CC C7(DB)
0550:A9 CC 8D F4 02 60 86 A5(D2)	0030:2B E4 D8 EE E3 F8 E6 F0<43>	08101A1 C7 EE C7 A6 C7 BD C7(5F)
0558:84 A6 20 1E 21 C0 00 30 <af></af>	8838:90 FØ 8A FØ 84 EA 43 EC(FD)	ØB18198 C7 CC C7 CF C7 BD C7(45)
0560:05 A6 A5 A4 A6 60 20 00<12>	0840:7C EC 15 F8 22 EC 75 EF<43>	
9568:E4 53 74 6F 70 70 65 64(00)	0848:97 EF 20 00 E4 53 78 65 <f3></f3>	0820:00 00 C2 C8 98 C7 98 C7 <df></df>
0570:00 4C 9D C8 A8 FD A2 86(21)		8828:89 C8 89 C8 89 C8 89 C8<86>
9578:4C FE 20 A0 27 A9 80 99(98)	8858169 63 68 65 72 28 76 6F<30>	0830:00 00 23 C8 CF C7 CF C7(1D)
0580:99 21 88 10 FA 68 8A 48(3F)	8858:6C 6C 88 4C 9D C8 85 A4<8E>	0838:00 00 00 00 00 00 00 00 00<4E>
0500177 21 00 10 FM 00 BM 48(3F)	0868:18 65 8A A6 A5 BB 69 80(6B)	8849:00 00 00 00 00 20 20 20 37>
0588:98 48 AZ 10 A9 87 9D 42<82>	8860:CC E5 02 ED E6 02 B0 DA(17)	@B48: 20 20 20 2C 28 22 22 28<73>
0590:03 A9 80 9D 48 83 9D 49 <e2></e2>	0870:38 A5 9A F5 00 85 A9 A5<70>	0950:1E 1A 1C 32 04 00 00 30(E2)
0598:03 20 88 C4 85 9D E6 95(7A)	0978:88 F5 81 85 AA 18 85 80(88)	0858:30 30 30 30 30 2E 2E 32(E3)
05A0:68 A8 68 AA A5 90 66 86<56>	0980:85 AS 85 AB 45 A4 85 A7(59)	ØB6Ø: 32 32 32 32 Ø4 32 32 32<62>
05A819D A6 97 E6 97 BD 80 64 <aa></aa>	0989:85 81 85 A6 85 AC 67 00<52>	
90 85 9D 68 A2 FF 9A(78)	0890:85 AB 85 00 65 A4 95 00(C6)	0B68:32 32 32 32 32 32 32 32(FD)
9588:4C 35 CB A9 88 85 95 28(19)	00000-05 Mt 10 00 05 M4 43 66(C9)	@B70:32 32 32 32 32 32 32 32(84)
0500127 C1 85 92 20 27 C1 85(FC)	8898:85 81 69 90 95 81 E8 EB <d8></d8>	ØB78:32 32 32 32 32 3A 2A(74)
85CB: 93 AB 30 E9 20 27 C1 B5<3C>	88A8:E8 8C 98 EE A6 AA 18 BA<82>	0880:32 32 2A 32 32 28 32 32(4D)
	88A8:65 A6 85 A6 18 8A 65 A8<25>	@B88:32 28 00 32 32 32 32 32(95)
0500:94 A9 04 A2 88 20 FF C3(EF)	0880:85 A8 E8 A4 A7 F8 25 88<38>	@890:32 32 00 32 32 32 00 00(81)
05DB:A5 92 91 AB CB A5 93 91 <e1></e1>	8888:81 A5 91 A7 98 00 F8 F8 <c2></c2>	8898:00 00 00 00 00 00 00 00 00 AE>
05F0:AB C8 A5 80 91 AB C8 A5<74>	98C0:18 C6 A6 C6 A8 88 81 A5(4C)	@BA8: @0 @0 @0 20 20 20 20 20 (9A)
05E8:81 91 AB A5 92 A4 93 20(59)	2808:91 A7 88 B1 A5 91 A7 88(29)	0BA8: 20 2C 28 22 22 28 10 1A <fa></fa>
05F012C C2 A9 00 85 8F 20 AE(82)	8808:81 A5 91 A7 B8 B1 A5 91(EA)	
05FB:C4 A5 95 C5 94 90 F7 A5(EE)	08D8: A7 98 D8 E9 CA D8 E2 A8(BC)	0BB0:10 02 30 01 01 30 30 30 (00)
0600:8F F0 19 A0 01 81 8E 48<2E>		30 30 2D 2D 02 02 02(43>
0608:A5 80 91 BE CB B1 BE AA(90)	08E8:00 68 A7 9C 9D 42 83 20(CE)	MILESTER 02 03 02 02 02 02 02 02(F6)
0610:A5 81 91 8E 86 8F 68 85 <ce></ce>	09E8:AF 28 C8 08 38 81 66 98(20)	8BC8:02 82 02 02 02 02 02 02 02(DE)
8618:8E 4C A0 C1 60 85 D4 84 CB>	08F0:48 28 EF E4 49 2F 4F 28 (BD)	0800:02 02 02 02 02 02 02 02 02 <ea></ea>
	08F8:45 72 72 6F 72 28 20 20<06>	0808:02 02 02 02 2A 2A 02 32<00>
0628:05 A9 00 85 06 85 07 85(14)	8992:88 A8 88 48 28 E8 C1 28 <f2></f2>	WHENTYN 02 02 28 02 32 32 28(32)
8628: DB A2 10 FB 86 D4 26 D5(AA)	8988:EA E4 4C 9D C8 28 27 C1 <f7></f7>	@BEB: 00 32 02 02 32 32 32 32 32(9C)
06 65 06 85 06 A5 07(89)	8918185 96 A8 88 84 9A 84 A3<8C>	08F0:00 02 32 32 A7 80 65 98 <d8></d8>
0638:65 D7 95 D7 A5 D9 65 D8 <f8></f8>	8918:28 27 C1 99 88 84 C8 A6(C7)	88F8: D8 82 46 98 98 4C F3 E9 <fe></fe>
0640:85 D8 CA D0 E7 D8 60 20(50)	8928:95 E4 96 98 F3 C9 14 D8<54>	8098:A9 83 20 F3 E9 20 90 C7<8A>
0648:8E C1 A9 90 85 D9 A5 D8(48)	0928:05 A9 16 99 7F 04 99 80(F0)	
0650:20 DC C2 A5 D7 20 03 C2(F1)	8939:84 CS D2 FA CS B4 97 AD<48>	0C08: A9 84 46 98 4C F3 E9 A9(C1)
0658:A5 D6 20 03 C2 A5 D9 F0(AA)	9030-01 B4 D5 DD DD DD DD DA DE-CA-	0C10:83 20 F3 E9 A9 97 20 F3 <c5></c5>
8668:12 68 48 4A 4A 4A 4A 28 <zb></zb>	8938:81 84 85 9D AD 88 84 85 <f6> 8948:99 8A AB B9 21 C3 85 D4&lt;82&gt;</f6>	BC18:E9 4C A6 C7 46 98 A9 B5<3F>
8668:8C C2 68 29 0F D0 84 A4(EE)	0740177 OH HO BY 21 US BS U4(82)	MAN 6A CB 28 BD C7 A9 88 CE>
0670:D9 F0 17 09 30 85 D9 48(E9)	8948:89 22 C3 85 D5 20 88 C5 <ec></ec>	F3 E9 A9 80 20 A9 00(E0)
0678:8A 48 98 48 BA BC 03 01<07>	0958: A5 9A F0 12 A2 80 20 FF<8C>	0C30:85 98 A6 98 F0 14 BD FF<67>
MACON A4 A2 20 Et CT 40 BA BC 83 01(07)	8958: C3 A6 9A B9 88 D9 91 AB<67>	0C38:D7 C7 8F 90 08 C9 93 90 <f3></f3>
2680: A6 A2 20 FE 20 68 A8 68(86)	09601CB CA D0 F7 86 9A 60 6C(F9)	0C40:09 C9 95 F0 05 A9 B7 20<9C>
06881AA 68 60 20 BE C1 A0 16(11)	0968: D4 00 20 41 C5 C6 9B C6 <b4></b4>	9C49:F3 E9 4C 9D C7 A9 83 85(CC)
0690:A5 D8 Z0 46 C2 A5 D7 20<3C>	0970:98 F0 28 A4 98 89 81 D8<34>	8C50:98 4C 6A CB A9 83 85 98 <cb></cb>
Ø678:3D C2 A5 D6 48 4A 4A 4A(27)	2978:48 B9 80 D8 18 49 88 48(14)	B: F3 E9 A9 97 4C 4A CB(27)
86A8:4A 28 46 C2 68 29 8F 89(B5)	8788: A9 BF 29 F3 E9 28 45 C5<55>	BC60: 20 9D C7 46 98 A9 88 4C469>
06AB:18 99 99 21 CB 60 3A 65 <c7></c7>	0988:48 26 F3 E9 68 28 F3 E9<56>	0000:20 70 C7 46 78 N7 88 4C(69)
0680:69 6C 65 AD 04 B9 4F C2(54)	000000 20 F3 E7 00 20 F3 E7(36)	BC68:F3 E9 A9 B3 28 F3 E9 A9<39>
2688:99 A7 21 88 10 F7 A9 87 (CE)	0990:4C 31 C9 AD 81 D8 48 A5<95>	0C70:9F A4 A0 20 6A C8 A9 00<13>
GACG-AS IN OD AS OF AS OF STATE	0998:90 18 69 73 48 4C 26 C5<38>	BC78:85 AB 85 98 A9 84 4C F3(64)
06C0:A2 10 9D 42 03 A9 0E 9D <ed></ed>	89A8: A8 88 84 98 A8 88 84 9C<6C>	0000:E9 60 46 98 C6 A0 F0 06 <fb></fb>
06CB:48 03 A9 00 9D 49 03 A9<83>	09A8:84 A0 A9 11 BD 00 DA 20<4B>	9C88:C6 A0 A9 83 D0 0F A9 87<94>
06D0:D4 9D 44 03 A9 00 9D 45(AC)	0980:00 C5 90 FB 85 9E AA 80(03)	0070:20 F3 E9 A9 B9 20 F3 E9<00>
0608:03 20 88 C4 4C 80 C2 A5<46>	8988:C9 C6 85 9F A4 9C BE 88<56>	0098:A9 00 20 F3 E9 20 6A CB(AD)
06E0:04 05 05 F0 1A 20 03 E4 <ed< td=""><td>89C8: DA BD 27 C7 C5 9F 98 1C&lt;24&gt;</td><td>OCAO: A9 84 4C F3 E9 A9 AB 2C&lt;52&gt;</td></ed<>	89C8: DA BD 27 C7 C5 9F 98 1C<24>	OCAO: A9 84 4C F3 E9 A9 AB 2C<52>
06E014B 65 69 6E 20 42 61 73(9D)	89C8: AA DØ 81 68 89 80 DA C6<11>	BCAB: A9 9B ZC A9 9A 1B 65 AB< 93>
06F0:69 63 20 50 72 6F 67 72(F0)	09D0:9C AB DA AA BD FD C5 B5<60>	
86F8:61 6D 6D 88 4C 9D CM A5<78>	0708:04 BD FE C5 85 05 20 08<08>	0CB0:65 A0 48 A9 83 20 F3 E9<5A>
0700:D6 C5 DA A5 D7 E5 DB B0(E8)	09E0:C5 4C 5D C5 A5 9E C9 3C(D3)	BCBB: A6 9C C6 9C BC 00 DA B9<57>
0708:08 20 27 C1 E6 D6 D0 EF <a9></a9>		@CC0:00 DB 85 78 68 A2 00 86 <fd></fd>
0710:E6 D7 D0 EB A5 DA C5 DC<73>	09E8: D0 2E 89 00 DA 88 C9 58<60>	8CCB: A8 28 F3 E9 4C 9D C7 A8(A4)
GATELOS DE CE DE CE DE CE	09F0:F8 22 C9 6B F0 1E C9 38(64)	0CD0:00 64 98 20 45 C5 4C 31(3E)
0718:A5 DB E5 DD B0 ZD 20 27 <b7></b7>	09F8:F0 10 C9 37 F0 0C C9 39<1E>	BCD8: C9 AØ ØØ 84 98 20 45 C5(5A)
0720:C1 AB Z0 27 C1 AA 98 09 1B>	8A88:F8 88 C9 38 F8 84 C9 3F<3C>	0CE0: A9 B3 20 F3 E9 4C 31 C9<03>
0728:3F 9D 00 DB 20 27 C1 20<26>	BARB: DO EB A9 B3 20 F3 E9 A9(D1)	9CEB: 98 48 AØ 8Ø 20 CF C7 A4 <cb></cb>
0730:27 C1 20 27 C1 20 27 C1<16>	8A18:8F 28 F3 E9 A5 9E A4 9C(69)	OCFU: A3 68 38 E9 66 99 B1 05<29>
0738:20 27 C1 20 27 C1 18 A5<68>	BA18:C9 37 98 17 C9 3C B0 13(6B)	QCF8:F0 02 A9 40 99 80 85 A9<2F>
8740:DA 69 08 85 DA 90 CD E6<3F>	8A20:09 3A F0 0F A6 9B BD FF<26>	0000:00 99 92 05 99 83 05 99(5E)
		(3E) // CO 77 CO 77 CO 26)
	-	

Listing 1. Turbo-Basic-Compiler (Fortsetzung)

@D@8:84 @5 99 85 @5 18 98 69<41>	@FE8: E4 29 7F 20 18 C2 BD BA(41)	1208: A5 D5 4C A9 E9 A6 D4 4C<62>
9D10:06 85 A3 69 7A BB 20 F3:68>	0FF0:E4 10 F2 A5 A4 F0 18 20(C9)	1200:36 EB 68 68 8A 48 88 88(EZ)
0D18:E9 28 A9 85 69 88 4C F3(E4)	@FF8: 83 E4 28 6E 69 63 68 74<95>	1208: 28 FC E8 68 4C 84 E9 A2 <ef></ef>
0D20:E9 A0 48 4C CF C7 E6 A0(51)		
	1000:20 63 6F 6D 70 69 6C 69(F0)	12E8:55 28 FD E6 E6 D4 DØ D5(A7)
BD28:68 68 64 9C 4C 6D C5 B9 <e1></e1>	1808:65 72 62 61 72 80 60 A5(B1>	12E8:E6 05 4C 1C E7 A9 17 20<26>
0D 30:02 DB 99 90 D8 CB C4 98<76>	1010:99 A0 00 28 E8 C1 20 03\6C>	12F0: A9 E9 38 A5 86 E5 84 08<45>
0D38:D0 F5 C6 98 4C F1 C8 8A(28)	1018:E4 20 2A 2A 2A 20 4D 55 <d9></d9>	12F8:20 A9 E9 A9 A0 20 A9 E9 <c5></c5>
0040:99 00 D8 C8 C8 R9 01 D8(20)	1020:45 4C 4C 20 2A 2A 2A 00 BE>	1300:28 AS 87 ES 85 20 A9 E9(89)
0D48:99 00 D8 C8 C4 98 D0 F5<47>	1028:4C 90 C0 04 4C 49 53 D4(C7)	1308:C8 89 00 D6 85 E0 C8 89 <ba></ba>
0050:C6 9B 4C 3B C9 20 20 CB(09)	1030:85 45 4E 54 45 DZ 0F 43(F5)	
		1310:00 DB 85 E1 98 48 A0 DD DC>
DD58:98 23 48 20 80 E4 49 6E<5D>	1038:4F 4E D4 16 4E 45 D7 18(74>	1318:81 E0 18 69 01 85 E2 A2 <ac></ac>
0D60:74 65 67 65 72 20 3C 30<3D>	1040:4C 4F 41 C4 19 53 41 56(18)	1320:86 20 FF C3 A0 00 B1 E0<60>
0D6B: 20 6F 64 65 72 20 3E 36<07>	1048:C5 34 43 53 41 56 C5 35<73>	1328:91 AB CB C4 E2 D0 F7 68(3A)
0070:35 35 33 35 00 68 A8 A9(C6)	1058:43 4C 4F 41 C4 37 45 52<68>	1330:AB A9 10 4C 21 E5 A2 55(B1)
2078:00 85 04 85 05 A9 89 99(AC)	1058:52 4F D2 53 2A CC 55 52(D1)	1338:20 A2 88 20 A2 8A 20 A2 8B>
ØD80:00 D8 A5 D4 99 01 D8 A5<87>	1860:45 4E 55 CD 56 44 45 CE(ZE)	1340:8C 20 A7 E7 E8 A9 84 2C(8C)
0D88: D5 99 02 D8 C8 4C E4 C8(34)	1968:57 44 55 4D DB 58 54 52<4A>	
		1348:A9 85 4C 36 EB A2 09 2C<20>
0090:A9 00 20 F3 E9 C6 98 A0(C6)	1070:41 43 C5 00 54 4F 48 45<35>	1350:A2 29 2C A2 49 20 08 E9(F3)
0078:00 89 00 08 10 67 C9 88(D5>	1078:4E 20 A3 20 15 C1 A5 92<84>	13581A9 AA 20 A9 E9 A9 9B 20<35>
00A0:80 62 C9 AF F0 5C 80 58(E8)	1080:A4 93 20 EB C1 A9 3A 2C<67>	1360: A9 E9 20 08 E9 A9 A8 20<75>
2DA8: C9 96 F2 58 C7 81 F0 53(83)	1088: A7 20 2C A9 78 4C 18 C2<4D>	1368: A9 E9 A9 8A 4C A9 E9 00(32)
8088: C9 8C F8 4F C9 8E F8 4B(AC)	1898:68 85 D4 68 85 D5 66 D4(1E)	1378:81 82 83 FF FF FF FF FF (08)
QDBB; C9 94 FØ 47 BE 81 D8 C9 <f9></f9>	1098:00 02 E6 D5 A0 00 B1 D4<7F>	1378: FF FF FF FF FF FF FF FF(9E)
0DC0:9A F0 50 C9 9C F0 57 C9<51>	19A9:F8 66 28 18 C2 4C F5 E4 <cc></cc>	1380: FF FF FF FF FF FF FF FF FF A6>
QDC8198 BQ 39 C9 80 FB 3C C9<39>	10A8: A5 D5 48 A5 D4 48 68 98< BA>	1388:FF FF FF FF 1D 1E 1F 20<7D>
0D00:82 FB 4E C9 83 F0 56 C9(55)	1080:48 BA AØ ØØ 20 E8 C1 20(91)	1390:21 22 23 24 25 26 27 28(AA)
0DD8:84 F0 39 C9 88 F0 5C C9(EE)	1088:00 E4 51 4A 53 42 00 68(98)	1398:29 2A FF FF FF FF 2F 30(2F)
ØDE 0: 89 FØ 63 C9 BA FØ 10 C9<23>	1000:AB 68 8A AA A9 28 28 A9<1F>	13A0: 31 32 33 34 FF 36 FF FF 663>
ODEB: 88 FØ 20 C9 80 FØ 15 C9(80)	10C8:E9 BD 80 DF 20 A9 E9 BD <ae></ae>	13A8:FF FF FF FF 3D 3E FF D3 <bd></bd>
0DF0:86 F0 10 C9 93 F0 8C C9(9D)		1380:41 D0 D1 44 45 46 47 48(DF)
	1000:01 OF F0 DB 4C A9 E9 20(CB)	
9DFB:87 F8 6F C9 85 D8 86 4C(88>	1808:00 E4 43 6F 6D 70 69 6C(AD)	1388:49 4A 4B 4C 4D 4E 4F 50(45)
0E0013B CA C8 C8 C8 C8 C4 98(29)	10E0:65 72 28 46 65 68 6C 65<76>	1300:51 52 53 54 55 56 57 58(40)
0E08:90 8F 60 89 03 D8 C9 83(33)	10E8:72 00 4C 9D C0 A4 9B C0(16)	1308:59 5A 5B 5C 5D 3E 5F 35(2A)
0E10:00 03 4C F6 C8 4C CB CA(88)	19F0:07 D8 25 AD 00 D8 C9 82(68)	1300:37 FF FF 38 39 FF FF FF(F7)
0E18:4C 05 CB A2 A9 2C A2 AA<02>	10F8: D0 1E AD 02 D8 C9 88 D0 82>	1708:FF FF 3A 6B 6C FF FF FF(78)
0E70:7C A2 8A 89 02 D8 C9 83(38)	1100:17 AD 05 D8 C9 A0 D0 10 <af></af>	
		13E0: FF
0E28: DØ DA 4C E0 C8 EØ 84 DØ(C2)	1109:AD 01 DE CD 86 DE DO 08(E1)	13E8: FF FF FF FF FF CO(CF)
0E30: D4 4C D0 C8 E8 83 FB F9<71>	1110:80 95 DB AG 92 4C C6 E5(ED)	13F9:C2 86 2C 2D 2R C4 00 CD<70>
ØE38:40 SE CA EØ 83 FØ 04 EØ(CC)	1118:A0 00 C4 7B F0 41 89 00<2E>	13F0:C3 80 C1 C6 B7 C7 2E C9 <e9></e9>
0E40:B4 D0 C2 4C E6 CB B9 03(D1)	1120: D8 AA BD CE E7 10 30 C9(BD)	1400:CA CB C5 C0 CC D2 CD FF<53>
0E48: 08 C9 46 F0 08 C9 97 F0(E2)	1128: C0 90 1B C9 D7 00 A8 29(FF)	1400: FF 89 8A 8B 8C FF 8E BF(CD)
005010F 09 55 00 AE A9 93 20(3F>	1130:3F 9A AA BD CE EB 85 D4(68)	1410:90 91 92 93 94 95 96 97 <a5></a5>
0E58:A9 86 99 00 08 4C 2D C9(80)	1138:BD CF E8 83 D5 B9 00 D8<17>	1418:98 99 04 FF 9C FF CF CE<21>
0E68:A9 96 99 08 08 C8 C8 4C(46>	1146:20 08 C5 4C 89 E5 A9 A2(16)	1420: CE CE CE CE CE CE CE CE<70>
20E 68: D/2 CB E2 60 F3 0E E2 82 CC>	1148:28 A7 E7 C8 57 08 D8 28(E5)	1428: CE CE 05 06 FF FF FF FF(93>
0F70:F0 19 E0 89 D0 03 4C A8<28>	1150:A9 E9 BD CE E7 29 3F 20<0F>	1430: FF FF FF FF FF FF FF FF FF (58)
ØE 7B: CA 4C A6 C9 B9 Ø4 D8 C9<6C>	1158:21 E5 C8 C4 98 D0 BF 60(12)	1438: FF FF FF FF FF FF FF FF (60)
0E80:85 D0 F6 A9 88 20 49 CA <dc></dc>	1160:38 E9 58 2C A9 88 2C A9(CA)	1440: FF FF FF FF FF FF FF FF (68)
0008:40 20 C9 B9 03 08 C9 85(4A)	1168:81 2C A9 85 2C A9 84 85<50>	1448: FF FF FF FF FF FF FF FF FF<70>
0E90:D0 E7 A9 BD 20 49 CA 4C<43>	1170:EB CB B9 00 DB B5 E6 CB<34>	1450:FF FF FF FF FF FF FF FF(78)
0E98:E4 CB 8A 38 E9 24 C9 84(BA)	1178:89 00 08 85 E7 84 E9 40<83>	1450:FF FF FF FF FF FF FF FF<00>
0EA0:10 07 99 00 08 4C E5 CB(FF)	1180:85 B1 E4 99 E0 00 88 10<87>	1460:FF FF FF FF FF FF FF FF (88)
0EAB: 79 00 08 98 48 89 02 08 86 >	1188:FB AS EB DB B6 AS E8 C9<28>	1468:FF FF FF FF FF FF CC(5D)
ØEBØ: 99 Ø1 D8 C8 C4 98 DØ F5 <f6></f6>	1198:04 F0 10 A5 E5 05 E4 05<15>	1470:E5 C9 E5 4C E7 FC EB F1 <cb></cb>
ØEBB:C6 98 68 AB 68 EØ AØ FØ\71>	1198:E3 05 E2 D0 34 A5 E0 C9(5F)	1478: C6 3E E7 AF E7 AC E7 B2<60>
0EC0:13 E0 A2 F0 12 E0 A3 F0 D6>	11A0:40 D0 2E A5 E8 A4 E9 A6<2C>	1480: E7 98 E7 98 E7 9E E7 95(C1)
WEC8: 11 E0 B8 F0 15 E0 B9 F0<09>	1148:E1 C9 66 F8 22 C9 67 F0<2A>	1488: E7 BE E5 BF E5 C3 E5 1E<46>
@EDØ: 11 4C A6 C9 A9 A6 2C A9<8E>	1180:36 C9 68 F8 4C C9 69 F8<72>	1490:E9 21 E9 85 E9 2E E9 38(EC)
0E08:A7 2C A9 AB 99 00 DB 4C(AB)	1188:4F C9 6A FØ 32 C9 04 FØ(4D)	1498:E9 4E E9 51 E9 20 05 E9<75>
0EE0:E5 C8 B9 04 D8 C9 83 D0.99>	11C0:36 C9 05 F0 46 C9 00 F0<1C>	14A0: A9 A0 2C A9 A2 2C A9 A9(CD)
DEE8:E8 C8 20 20 CB 88 80 E1<1D>	11C8:51 C9 81 F0 4A D0 54 EB<8E>	14A8: 24 8A 20 A9 E9 C8 B9 00(AD)
@EFØ: A5 D5 D0 DD 18 B9 01 D8<50>	1100:02 D0 50 A9 71 20 A9 70<82>	1480:D8 4C A9 E9 A9 87 20 89<26>
0EF8: 69 03 99 00 DB AS D4 99(50)	11D8:2C A9 AF 2C A9 72 2C A9<9B>	14881E9 4C 0C E9 A2 BA 2C A2<6E>
0F00:01 D8 C8 C8 4C F2 CA 89(82)	11E0:43 2C A9 42 4C 21 E5 E0<70>	1400:06 20 A2 D4 A9 A5 28 36(A6)
0F08:04 D8 C9 56 F0 08 C9 57(32)	11EB:02 FO EB A9 73 DO 1E EO(EC>	14C8:EB E8 A9 A4 4C 36 EB 28<08>
	11F8:02 FD E9 A9 76 D0 16 E0(AD)	14D0:F6 E6 A9 B1 A2 D4 4C 36(C6)
0F10:F0 0A C9 5A D0 80 A9 94<86>		
0F18:2C A9 8C 2C A9 8E 99 01<9E>	11F8:01 F0 E4 90 E5 A9 40 D0-(34>	14D8:EB A9 A2 C8 BE 20 D8 20(86)
0F20:08 A9 83 99 80 DB C8 4C(79>	1200:0C E0 01 F0 D4 A9 74 2C<9F>	14E0:36 EB 80 00 DB 10 03 A9(3B)
DF28:20 C9 C9 49 FB 23 C9 48(A2)	1208:A9 75 2C A9 79 48 A9 A9(BD)	14E8:3C 2C A9 1A 4C 21 E5 A9(41)
0F30:F0 1F C9 A0 F0 41 C9 36<6D>	1210:20 36 EB 68 4C 21 E5 A9(F7)	14F0:6D 2C A9 6E 48 A9 A2 20<08>
0F38: D0 27 B9 02 D8 85 D5 89<56>	1218:78 2C A9 77 48 28 82 E9<97>	14F8: A9 E9 C8 B9 00 D8 20 A9 <fc></fc>
0F40:01 D8 85 D4 A2 00 A1 D4 <cc></cc>	1220:4C 6D E6 A5 82 85 D4 C3(2B)	1500:E7 68 4C 21 E5 A0 00 2C(05)
	1228:84 A5 83 85 D5 E5 85 B0<00>	1508: AD 06 AP AP 20 A5 E7 AD 4C>
0F48:F0 04 49 80 81 D4 4C 2D:83>		
0F50:C9 B9 03 D8 99 00 D8 C8-28>	1230:21 A0 05 B1 D4 D9 E0 00<42>	1510:F8 4C 3D EB A9 1F 2C A9 <ac></ac>
0F58:C4 9B D0 F5 C6 9B 4C DB<9C>	1238:00 05 88 10 F6 30 24 18(64>	1518:10 2C A9 1C 2C A9 03 2C(EC)
0F60:C8 4C A4 C9 B9 03 D8 C9:C2>	1240:A5 D4 69 06 85 D4 90 02 <d7></d7>	1520:A9 02 2C A9 00 0A AA BC(56)
0F68:10 90 F6 C9 28 80 F2 69(D1)	1248:E6 05 C5 84 A5 D5 E5 85<2C>	1529:01 DE DO 10 8A 20 EB C1 <a4></a4>
0F 70: 93 99 00 08 4C 2D C9 A9<80>	1250:98 OF A9 86 A2 84 28 FF <cb></cb>	1538:20 E7 E4 20 00 E4 51 4A(1E)
0F78: AF 99 00 08 4C 20 C9 98(52)	1258:C3 A0 05 89 E0 00 91 D4 <e6></e6>	1538:53 41 00 68 BD 00 DE AA(25)
I		
0F80:48 C0 09 00 D8 85 E0 C8<64>	1260:80 10 FB A9 0F 20 A9 E9(89>	1548: A9 28 28 A9 E9 8A 28 A9<6F>
0F88:89 00 00 05 61 A0 05 81(AA)	1268:38 AS D4 E5 82 88 28 A9<88>	15481E9 98 86 A4 A6 9A 9D 00(A8)
0F90:E0 99 D4 00 88 10 F9 20 <eb></eb>	1278: E9 A9 A8 20 A9 E9 28 A5 <af></af>	1550: D9 A6 A4 E6 9A F0 1B 60 4D>
0F98: 2A 21 68 AB 60 00 E4 27(70)	1278:05 ES 93 20 A9 E9 A4 E9(D3)	1558:20 00 E4 41 75 73 64 72 <c3></c3>
0FA0:F9 20 DA E4 E6 90 D0 82 <ad></ad>	1280: A5 E8 10 08 20 02 E9 A9 <a7></a7>	1568: 75 63 6B 28 7A 75 20 6B<8A>
0FAB:E6 91 68 85 D4 68 85 D5(31)	1288:38 C6 E8 10 02 A9 3F 4C(77)	1569:6F 6D 70 6C 65 78 00 4C(D3)
		15/0:90 C0 98 48 8A 48 A9 80 09>
0FR0:E6 D4 D8 02 E6 D5 A8 00 D6>	1290:21 E5 AZ 46 20 FD E6 A9 <b9></b9>	
ØFB8:81 D4 FØ Ø6 20 18 CZ 4C/B3>	1298:A8 A2 88 4C 36 EB CB B9<9F>	1578:A2 80 28 FF C3 A0 00 89<15>
ØFC0: ØF E4 20 EA E4 A5 D5 48(89)	12A0:00 D8 85 D4 C8 B9 00 D8(C9)	1580:00 D9 91 AB B9 80 D9 99 DC>
0FC8: A5 D4 48 68 20 DA E4 AZ (0F)		1598:00 D9 C8 10 F2 84 9A 68(10)
	12A8:85 D5 C9 C0 90 0C C9 D8(99>	
		1590: AA 68 A8 60 B6 A4 A6 98<59>
@FDØ:00 RD BA E4 B5 A4 F0 00<20>	1280:80 20 C9 D3 F0 1C C9 D0<4C>	
0FD0:00 RD BA E4 R5 A4 F0 00<2D> 0FD0:05 99 F0 09 E8 BD 8A E4<79>	1288:80 20 C9 D3 F0 1C C9 D0(4C) 1288:90 18 A9 A5 2C A9 A4 A6(1D)	1598:90 00 08 A6 A4 E6 98 FØK 40>
@FDØ:00 RD BA E4 B5 A4 F0 00<20>	1280:80 20 C9 D3 F0 1C C9 D0<4C>	

Listing 1. Turbo-Basic-Compiler (Fortsetzung)



15	AB: A9	20	A9	E9	AD	81	84	49<8F>	1	1888: AB	CB	A5	80	71	AB	CB	A5<24>	
15	99: 26	FØ	02	A9	PF	26	A9	E9<67>		1890:81	91	AB	618	38	A5	8A	E9<21>	
15	88:4C	30	EB	20	7A	C8	A4	9B<06>		1098:03	85	8A	BØ	82	C6	88	AØ<000>	
15	CØ: CØ	03	DØ	17	AD	66	DB	C9 <d5></d5>		18A0:00	81	8A	60	20	9C	ED	20(8F)	
	CB: 87		10	A5	79	C9	ØC.	DØ<49>	1	18A8: F3	23	FØ	FB	60	AB	82	B1<01>	
	DØ: Ø5		28	20	84		AØ	88<83>	1	1889:8A	85	AC	88	Bi	BA	85	AB<36>	
											A3	80	69	103	91	AB	CB(10)	
	DB: 4C	13	E9	20	4C	E3	A9	Ø9<5E>		1999:18						-		
	E0: A4	99	CO	80	00	02	A9	58<8B>		18C0: A5	81	69	66	91	AB	60	48<10>	
15	E8:40	84	E9	20	E5			C8<1A>		18CB: 29	05	EC	68	AA	ÇA	DØ	@D<1B>	
15	FØ: 4C	2D	EB	20	E5	EB	A2	02<07>		18DØ: 20	20	E4	44	4F	20	66	65(EB)	
15	F8:AØ	FD	4C	20	EB	20	70	C8<80>		1808:48	6C	74	80	60	CA	DØ	11<50>	
16	00:20	4C	E5	A5	9D	C9	12	FØ<10>		18E0:20	80	E4	52	45	50	45	41<04>	
	08: F4	60	20	<b>Ø</b> B	C5	4C	4C	E5<8F>		18EB: 54	20	66	65	68	6C	74	DØ<43>	
	10:09	12	40	B4	£9	A7	ØC	20 <fb></fb>		16FB: 60	20	88	E4	57	48	49	40<48>	
				_							28		65	68	6C	74	20<8B>	
	18:84		4C	88	EA	A5	90	C9<58>		1BFB: 45		66						
	20:16		ED	20	84	EC	A4	97 <a5></a5>		1900:60	20	8E	ED	A9	02	2C	A9<59>	
16	28: CØ	01	DØ	89	A5	9D	Ľ9	12<3E>		1908:01	48	20	86	ED	85	D4	6B(7E)	
16	30:D0	<b>B</b> 3	20	48	C1	A5	910	C9<2A>		1910:C5	D4	00	<b>B</b> 3	CB	81	48	AA< 90>	
16	38:16	FØ	64	C9	12	FB	71	C9<80>		1918:C8	B1	8A	AB	A9	4C	4C	A1(B2>	
16	40:15	FØ	72	20	70	CB	C9	3E<95>		1920: E9	AØ	02	81	SA	85	AC	88(B1)	
	48:00	07	20	AB	F1	A9	28	DØ <e1></e1>	1	1928:B1	BA	85	AB	4C	E9	EC	A9 <d7></d7>	
	50:48	C9	54	DØ	07	20	AB	F1<04>		1930:00	20	A9	FØ	48	20	78	C8 <d1></d1>	
					_					1938:C9	84	Dig	ØE	20	AB	F1	A9(F5)	
	58: A9		DØ	30	AA	7B		\$2 <c8></c8>			_	-	_			_		
	60 L DØ	20	AD	90	DB	C9	81	DØ<4C>	1	1940:84	AØ	D4	20	A5	E9	A9	85(RF)	
	68:25	AD	01	D8	85	D4	AD	@2 <d2></d2>		1948: DØ	95	20	4C	€5	A9	A5	A0<21>	
	70:08		D2	AØ	90	B1	D4	C9 <c1></c1>		1950: D4	20	A5	E4	68	AØ	62	4C<2B>	
16	78:01	DØ	50D		B1	D4		A9<0A>		19581A5	E9	A5	88	48	A5	91	48<43>	
16	BØ: A9	20	36	EB	9	28	DØ	11<1D>	1	1960: 20	BE	EĐ	20	F1	C4	A9	03(92>	
16	88:20	BØ	F1	40	94	EA	20	40<94>	-	1968:20	FE	ED	A7	03	20	Εî	EC<98>	
	90:E5	A9	ØD	24	98	20	02	A9<67>		1970:68	91	AB	68	88	91	AB	6054A>	
	98:0E		84	E9	4C	94	EA	A9<04>		1978: A5	BA	85	D4	A5	3B	85	DS(EB)	
	A0:10	20	84	E9	A5	97	C9	33<28>		1980: A0	00	38	AS	D4	E9	03	85(A1)	
		_	_	_		84	E9	68(69>									E0<3E>	
	A8: D0	85	A9	11	28		_			1988: D4	82	02	CA	05	B1			
	BØ: A9	ØF	20	84	E.9	46	9D	A4<6E>		1990:F1	C9	05	DØ	98	49	1F	20<40>	
	PB197	89	80	24	C9			E4 <f1></f1>		1998:84	E9	29	F1	C4	A9	99	20(02)	
16	C0: C7	15	FØ	04	C9	12	DØ	824.89>	1	19A0:E1	ЕĊ	4C	719	ED	A9	62	28 (CF)	
16	CB: 20	48	Cl	40	94	EA	BA	FØ <ae></ae>	1	19A8: 68	ED	20	F1	C4	20	F3	EC<4D>	
-16	DØ: 00	A9	8D	29	A9	E9	78	20<53>		1980:4C	88	ED	20	SE.	ED	A5	9D<58>	
16	DB: A9	69	BA	4C	A9	E9	A9	85<00>		1988:C9	18	Fø	13	28	F1	C4	A6<39>	
	E.0: 4C	A5	E9	A9	95	20	A9	90<3F>		1900:80	Α4	80	20	7B	ED	A5	80(A6>	
	EB: 48	20	78	CB	68	20	F3	E9(5E>	- [	1908:85	8C	A5	91	85	80	60	CD <d3></d3>	
	F0: 20	7E	CB	Α4	9B	CØ	87	DØ<80>	- 1	1900: 7F	05	FØ	10	20	7A	CB	A4 <dd></dd>	
_	F9:8A	. —	90	DB	C9	89	00	03(E2)		1908:9A	A9	FØ	99	FE	DB	A9	00<89>	
	00:CD	24	DB	60		45	EB	DØ<4E>	1	1980:20	13	E9	86	95	E4	94	BØ (79)	
					50										_			
	88: SA	AD	96	DB	FØ	63	20	D8<16>	1	19E8: 06	20	27	Ci	4C	42	EE	60(85)	
	10:EB		02	DB	<b>C</b> 9	CB	92	SC< 30>	1	19F0: 28	F1	C4	A6	BE	A4	BF	20<83>	
17	18:07	DB	88	16	C9	D3	FØ	12 <a3></a3>	-1	19F8: 7B	ED	A3	88	85	BE	A5	B1<6A>	
17	20:C9	DB	70	ØE.	AØ	04	20	05 <f3></f3>		1A00:85	8F	68	A5	ap	FØ	3A	A2<13>	
17	20:69	AE	02	DS	AC	01	DB	40<91>		1A08:00	A0	20	A9	4C	91	80	CB(3E)	
17	30:20	EB	20	4C	E5	A9	13	4C <bc></bc>		1A10: B1	BC.	48	18	8A	65	89	91 <a2></a2>	
17	38:04	E9	20	43	EB	00	32	AD(62)		1A18:8C	C8	Bi	ØC	AA	A5	81	69(42)	
17	40:02	DB	C9	CØ	98	ØC.	69	DB< 88>		1A20:00	91	8C	BA	80	68	85	BC(23>	
-	46: 80	27	C9	D3	FØ	23	C9	DØ <dc></dc>		1828:60	A5	80	FB	14	AØ	99	B1 <d6></d6>	
	50: 90	1F	AB	04	20	FC	EB	AE<81>		1A38: BC	FØ	ØE.	A2	03	20	68	EE<59>	
	50:02	DB	AC	01	DB	20	2D	EB <c7></c7>		1A38120	18	EE	A9	600	BD	20	D9<66>	
	60: CB										20	88	E4	49	46	20	66<72>	
			01	EB	A8	FØ	95	A9<44>		1840:60								
				EB		84	_	A5<75>		1848:65		6C	74	20	60	A5	92 <ba></ba>	
	70:E9							84<65>		1A501A4			32	EB			C1 <fc></fc>	
	781E9		60					60 <f4></f4>		1A58:20							DØ<4D>	
17	90:3E	32	35	35	90	68	20	7A<6F>		1860:F4	AS	9A	ED	02	D9	A2	82(30)	
17	68 : C8	A9	97	20	99	EC	4C	4C<2A>		1A69: 20	FF	C3	A6	9A	B9	20	D9<87>	
17	90:ES	A9	21	20	A9	19	20	A9 <e3></e3>		1A70:91	AB	CB	CA	DØ	F7	86	9A <bf></bf>	
	98:17				48			EB(F6)		1478160	20	45		A9	91	20	9B <a1></a1>	
	A0:F0			40	E5			B4<26>		1480:EC	20	74		24	98		05<92>	
	AB: E7			20				04<38>		1488: A9			F3	E9	20		E5<08>	
	PØ: 20					69				1A70: A7		4C	84	E9	28		C2<92>	
	88:84							20 <a5></a5>		1078:04		89		07			4C <cc></cc>	
	CØ: 04							91<38>		1AA0: E5		719	CB	68	A6		EB <d0></d0>	
				-				92<4A>		1AA8: 1A		48		63	20		C8<3B>	
	DØ: 20							F@ <dc></dc>		1AB0:20		E5			A9		20<42>	
	D8:86							A9<71>		1AB8: 36		A9	28				A9 <a7></a7>	
17	EØ: 27	4C	84	E9	20	45	EB	20 (BF >		1AC0:29		84	E9				@3<64>	
1.7	EB: 4C	E3	A9	18	40	84	E9	ADK2B>		1AC8: 20	36	EB	20	Fi	C4	A9	05<48>	
17	F0:81	64	C9	16	DØ	09	28	48 (E5)		1AD0:40	FΕ	ED	20	BD	FB	26	F1<28>	
17	F8:C1	28	E5	EB	A9	97	2C	A9(44)		1AD8: C4	20	86	ED	C9	05	DØ	13<13>	
	00:06		84	E9	20			A9<38>		1AC0: 20							B1 <a9></a9>	
	08:1E							C9<99>		LAES: AB							ZAKE D>	
	10:16			A9				F0<60>		1AF0:4C		E9		05			00(21)	
	18:A7			84				EC(4B)		1AF8:E4		4F		20			68(84)	
										1B00:60	74		60				20<52>	
	20:A9							B9(FB>										
	29:80							48(1E)		1808:E5					-		AS(BD)	
	30:C1		E2		49			A9<82>		1R10:9D							B4<5B>	
	38:0R		-	E9				8A<10>		1B18:EC		70					A9<77>	
	40:48					C9				1820:20	20	84			9D		16 <b6></b6>	
18	48:68	AA	60	A9	24	20	A7	25<65>		1828:00						CB	10<90>	
18	50:48	20	84	EC	20	45	EB	20<09>		1B30:00	ØE	A9	20	20	A9	4A	20<70>	
	58:4C		68					04<06>		1838: A9	28	48	20	84	EC	68	20<29>	
	60:2C			48				A9 <bb></bb>		1848: A9		48	20				48<4F>	
								20 <fc></fc>		1848:20		E9					DØ< 9A>	
	68:91													-				
				40	9.6	E.O	69	図3くりひき			6275	6.01	70	47	FR	20	4C< 099 >	
18	70:4C	E5	48					02<0D>		18501F2	86	60 2E		42 ES			4C<09>	
18	70:4C 78:AZ	E5 84	4C	FF	<b>C</b> 3	Α9	62	03<00> 2C<9E> 91<00>		1858:E5 1860: 20	A9	2F	DØ	E5	20	84	EC <fb></fb>	

1868: 20 A5 E9 A9 30 1970:20 84 EC 20 45 EB 20 4C<71> 1878: E5 A9 31 40 84 E9 20 84(BB) IRROSEC AD BI RA C9 RE 1DCAE> FØ A9 36 20 34<7F> 1888: A9 32 20 Α9 1890:48 20 BD FØ 68 48 98(91) 1878: 69 80 28 84 E9 A5 90 C9(2B) C8<445 TRAM: 14 DO FF AR AM 20 201 1BA8: 20 BØ F1 A5 9D C9 FØ<69> 15 1880:D7 DØ DB A9 39 20 3A(E3) 1989: AE B1 04 C9 16 D/A 85 A9(DA) 18C0:38 4C 84 69 48 20 F5 EB(B5) 1908: A0 BC 20 30 EB 68 20 A9<C4> 1800:38 48 20 E5 EB AØ 20(BD) 1908:30 EB 20 65 EB A0 9A 20(06) 1860: 35 FR 28 FS FR AS AC 84(07) 18E8: E9 AD B1 04 C9 36(82) 16 FØ 18F0: A9 3D 2C A9 54 48 AD B1<B0> C1(89) 1BF8:04 C9 1C DØ 21 28 48 1000:20 AA F0 AB AA A9 TO-FR(R4) 1008:84 F0 02 A9 37 E9(75) 28 85 1018: A9 AB AG 00 20 A5 E9 BA(89) 1018:40 84 E9 A9 30 48 20 7A(C4) 1020: CB 20 4E E5 68 20 AR 55< 76> 1028:40 84 EF 92<8E> FØ A5 20 A6 1030:99 00 DC A5 93 99 DD(66) 98 1038: 60 A9 च्या 20 84 6.0 20 AACROS 1040: FO AA A9 20(AC) 1F 40 A1 E9 1C48:48 C1 C9 00 D0 03 20 48(00) 1C50:C1 49 80 A8 20 48 C1 89(88) 1058:00 DR 85 98 98 A0 20 A6<9D> 1060: FO A9 A2 4C A5 E9 AD B1(87) 1068:04 09 16 DØ 05 A9 3F 40(20) 1C78:84 E9 A9 48 20 49 42 200180 1C78:A9 42 2C A9 43 44(03) 2C A9 1080:20 A9 45 20 A9 46 20 A9(E4) 1088:47 48 20 70 08 20 1090:68 40 84 69 20 65 40 ESKR1> EB 40(E9) 30 1098:00 20 EB A9 2C A9<48> 1CAD: 48 48 20 84 EC 20 E5 EB(AF) 1048: AØ 88 20 3D EB 28 E5 FRCF7 1CB0: A0 8A 20 4C FB(2D) 3D EB E9 02(08) 1088:20 70 CB C9 84 69 DØ 1CC0: C6 1CC0: A5 9B 20 4C E5 28 F1 C4<98> 9D C9 ØF 09 18(ZA) FØ 17 1CD0:F0 ØE C9 62 FØ 3A Ç9 6A<36> 1CD8:F0 33 4C 5F F5 42 40 20(37) ICE0: A2 4E 28 D0 01 E8 49 27<18> 1CE9: A0 00 28 A5 E9 88 20 84<00> 20 1CF01E9 76 **C8 A5** 98 B3<9E> 1CF8: D8 37 AD DI DS AC **B**2 DB(F5> 1000:28 A5 E9 EE 01 D9 9D<11> **A5** 1D08: C9 16 DØ ES 68 A2 50 20<08> 1D10:A2 52 28 00 01 28 AS E9 EB A9 2F(D9) 1D18:AR RA FIA 20 84(08) 1D20:E9 20 A6 FØ 20 A5 E9 EE (DF) 1028:01 09 A5 9D C9 16 DØ F1<AB> 1030: 6g 20 OID E4 4F AF 2E 474 B35 1038:4F 54 4F 47 4F 55KFE> 2F 53 65 1D40:42 20 78 70 80 4C 9D(FB) ACK 3A> 1048: CB C6 OR 20 31 C9 4C 1050:E5 20 4C 96 ES CA CA 9A(D7> 1050:C6 9A A9 57 4C 84 EP 28<45> 1060; EF E4 70 98 9B 9B 9B 20KFQ> 1D6B: 20 54 75 72 62 6F 20 42<A8> 1070:41 53 49 43 20 4C 20(96) 58 1078:43 6F 6D 79 65 1D80: 20 56 65 1D88: 20 31 2F 72 73 69 6F AFCACS 31 99 3D<AA> 20 20 1D90:3D 3D 3D 3D 3D 3D 3D< 66> 3D 3D(92) 1098:3D 3D 3D 30 30 3D 3D 1DA9:3D 30 3D 30 3D 30 3D 30<90> 3D 3D IDAG: 3D 30 3D 30 30<62> 3D 45 1DBØ: 3D SD 98 98 57 6C 63(D/) 1DB8:68 65 73 20 50 72 6F 67<CE> 1DC0:72 61 6D 6D 20 75 6C(59) 6F 6C(BB) 1 DC81 6C 20 63 6F 60 70 69 65 1000:69 65 72 74 20 72<64> 1008:64 65 6E 3F 9B 9B 2B 47(09) IDE0:65 62 65 AE 20 53 65<70> 69 1DEB: 20 76 75 73 65 20(E7) 1DF0: 64 69 65 20 4C 61 75 AA(78) 1DFB: 77 65 72 73 δE PD<62> δB 75 98 72 9B 1E00:60 65 20 1E08:69 73 63 1E10:20 75 6E 20 20 68 65 6E 31 (BA) 38 64 20 65<34> 20 75 1E18:69 6E 2C 6E 64 20(F5> 1E20:77 0B 68 6C 65 6E 20 64(8F) 1E 28: 61 73 9B 98 70 56 72 AF(10) 1E30:67 72 61 6D 6D 20 61<20> 64 1E38: &E &E 20 6D 69 74 20 64<12> 1E40:65 6E 20 50 66 65 69 6E(08)

Listing 1. Turbo-Basic-Compiler (Fortsetzung)

45.00								,						_			
1E48:74							9B<1D>		2128:00							20<35>	
1E'50:20	18	1E	2C	26	19	1F	2C <c1></c1>		2130:E6		AØ	62	71	AB		BA(A1)	
1E58: 20	18	1C	20	75	ЬE	64	20(2F)		2138:91	AB	88	A9	4C	91	AB	AØKE3>	
1E 60: 1E	1 D	2E	20	44	61	δE	6E<27>		2140:03	4C	15	F5	A9	80	2C	A9(98)	
18.68:20	52	45	54	55	52	4E	20<9F>		2148:FF	85	02	A9	A2	91	AB	C8(A8)	
1E70:64	72	BA.	63	69	65	ЬE	2E <dc></dc>		7150: B1	AB	85	D3		05		D2 <ff></ff>	
1E78:9B		9B	7F	43	6F	6E	74<68>		2158:30	-	20		Fé				
										96		20		4C		F5<40>	
1880:72		6C	20	44	20	2D	20<4C>		2160:91	AB	AA		B1	AB		E6<4E>	
1E88:52	ØA.	63	6B	6B	65	68	72(83>		2168:F5	ΑØ	26	91	AB	88	88	91 <f4></f4>	
1590:20	7A	75	6D	20	44	4F	53 <d4></d4>		2170:AB	18	A5	AB	69	02	85	AB SE>	
1E 9B: 98	98	7F	43	6F	6E	74	72<75>		2178:A5	AC	69	88	85	AC	C6	D3<78>	
1EAØ: 6F		20	52	0.0	3D	20	52(E0)		2180:00	D2	AØ	05	4C	1F		85(B1)	
IEA8: 65		6F	6F	74	98	88	20 (AE)		2188:D5	86	04	A5	86	85		A5<02>	
											_						
1E80:F7		29	7F	C9	24	E.S	07 <a2></a2>		2170:87	85	D7	AØ		B1		C5<93>	
1EBB: C9	12	DØ	96	4C	B4	F4	4C<9D>		2178:D4		B1	D6	E5	D5	90	09<4D>	
1EC0: D8	F4	C9	31	90	E9	C9	39(16)		21AØ:C8	CB	C8	DØ	FØ.	E6	D7	DØ<95>	
1E CB: BØ	65	80	A3	20	48	20	1CKCF>		21A8:EC	DØ	07	98	91	D6	C5	D4<78>	
1ED0:C1	A9	24	60	99	21	68	29(A5)		21BQ:F@	12	20	46	F6	20		E4<30>	
1ED9: 1F		9A	21	A9	10	BD	98<8D>	1	2188:5A	65	69	6C	65	20		65<7E>	
	AZ	19		83	C4	A9	Ø3<9A>										
1EE0:21			20	-					2100:68	4C	74	88	E8		B1	06434>	
1EE8: 9D	42	03	A9	A2	90	44	@3<47>			CB	81	06	60	B1	AB	A8<89>	
1EF 0: A9	20	90	45	03	A9	06	9D(EB)		2100:8E	99	DC	89	88	DD	10	AF(BD)	
1EFB: 4A	03	AP	20	90	48	23	20<67>		2108:20	44	F6	20	00	E4	4C	61<06>	
1F@0:88	C4-	20	A6	F4	A9	00	85-50/	1	2150:62	65	6C	29	66	65	68	6C<3D>	
1F08:92		03	85	8E	20	27	C1× 26		2128:74	99	60	A5	AB	A6	AC	85<97>	
1F10:C9		88	24	A2	ØD	20	27<9E>					_					
1F 18: C1	20								21F0:04	86	D5	A5	86	85	D6	A5<39>	
		18	C2	CA	DØ	F7	20<6D>		2168:87	85	D7	AØ	20	CS	CS	B1<53>	
1F 20 z 27	C1	C9	9B	08	F9	E.6	92<05>		2200:06		D4	C8	81	D6	E5	DS<83>	
1F29:A6	92	E0	3F	FØ	ØA	C6	BE<4E>		2208:BØ	07	C8	DØ	FØ	E <sub>6</sub>	D7	D0<90>	
1F30:00	DB	28	E7	E4	4C	68	F3: 612	1	2210:EC	28	A5	D6	E9	86	85	D9<88>	
1F38: A9		83	54		EA	E4	A9 <ca></ca>			02	C6	D7	B1	D6	85	93<57>	
1F40:17	85	54	A2	10	20	83	041631		2220:88		06	85	92	60	A9	A9<58>	
		85	93	A5	92	00	03/23/	1									
1F48:A9									2228:91	AB	CB	81	AB	AA	BD	00 <dc></dc>	
1F50:4C	BE		18	A5	58	69	7B< 95>		22301 DC	91	AB	AB	03	BD	90	DD <de></de>	
1F58:85	AB	A5	59	69	00	85	AC<461	1	2238:91	AB	1.0	62	28	37	F6	AB<48>	
1F 60: A9	83	85	ΘE	A6	93	FØ	10(21)	1	2240104	4C	1F	F5	20	F3	EC	10<81>	
1F 68: 18	A5	AB	69	ØD	85	AB	98 (25)	Į.	2249:03	4C	1F	F7	48	AG	02	B1 <ea></ea>	
1F70:02		AC	C6	8E	DØ	AB	A9. BR.	1	2750:8A	AA	88	B1	BA	20	4E	F6(C7)	
1F78:03	85		E6	AB	DØ	02	Ed FO	L		FO			81				
									2258:60		TD	C9		FØ	2D	C9 <f8></f8>	
1F 80: AC	CA		E4	AØ	@C	B1	AB BA	1	2260:02	FØ	3A	C9	92	FØ	40	28 <fd></fd>	
1F88:09	80	91	AB	88	1.0	F7	201 DA >		2268:00	€4	4E	45	58	54	20	66<9C>	
1F90:F7	CØ	A6	93	C9	9B	F0	49. 79	1	2270:65	68	6C	74	99	4C	A3	F6<72>	
1F98:C9	10	FØ.	27	C9	2D	FØ	23<70>		2278:20	96	E4	45	38	49	54	20<44>	
1FA0: C9	1.D	FØ	34	C9	3D	FØ	30-16/		228016F	68	6E	65	20	2E	2E	2E<9A>	
1FAB: C9		FØ	19	C9	28	FØ	15<4A>	1		4C		FA	20	00			
									2288:00							4C <ab></ab>	
1FP0:C9	1F	FØ	26	C9	ZA.	FB	22<60>		2290:4F	4F	50	20	66	65	68	6C(1D)	
1FB8:C9	31	70	D3	C9	39	BO	CF<42>	l	2298: 74	22	4C	A3	F6	20	90	E4<4E>	
1FC0:40	29	F3	CA	CA	CA	30	CZ <ba></ba>	1	22A@: 55	4E	54	49	4C	20	66	65 <c6></c6>	
1FC8:86	93	AØ.	ØC.	BI	AB	29	7F < E8 >	1	2248: 68	6C	74	22	4C	A3	F6	20<27>	
1FD0191	AB	88	10	F7	40	B2	F3<00>		2280:00	E4	57	45	4E	44	20	66<52>	
IFDB:EB		€8	E4	92	BØ	80	90(89)	l l				74					
									22B8:65	68	9C		20	4C		F6<4A>	
1FE0:E7		44	90	80	04	AD	A34 45 >	1	22C0: A6	BD.	FØ	23	A5	90		4E<00>	
1FE8:20	BD	81	04	49	29	85	82<7A>	Į.	22C8:F6	20	99	E4	45	4E	44	49<7E>	
1FF0:04	A2	03	AØ	81	B1	AB	29<3C>		2200:46	20	66	65	68	6C	74	<b>200/00</b>	
1FF8:7F	FØ	07	18	69	20	9D	90< D7 >	1	22D8: A9	02	B1	SC.	AA	88	81	BC<28>	
2000:04	E8	C8	CØ	20	FØ.	DC	CØ(69)		2260:85	ac	86	BD	BA	DØ	DD	88<18>	
2008:09	DØ	EA		2E	90	80	04<91>			BD	FB	A9	90	83	A2	20<6E>	
		E2			9D			1				46					
2010+E0						80	Ø4 <c3></c3>		22F0: EF	E4						6EKAC>	
2018:08								1	22FB161							72 <f2></f2>	
2020:38	E9	20	9D	98	21	CA	DØ<29>		2300:20	64	61	73	20			6D<97>	
2028:F4	AZ	10	A9	63	90	42	<b>0</b> 3<67>		2398:79	69	6C	69	63	72	74	65<81>	
2030: A9	SHR	90	44	83	A9	04	90:48		2310:20	50	72	6F	67	72	61	6D <c9></c9>	
2038:45				90			A9< 75>		2318:6D			98		A9	31	85(08)	
2040:00						-	A9 <d1></d1>						9C	7F		44<07>	
						_						_	18			EF<65>	
2048:70							20:00:		2328199		A7						
2020: EA							E4<28>		2330; E4		2E		_	42		1E(E2)	
2058: 7F				CZ	CF		D4<8B>		2228: 1E		80		86	-	A5	A9<33>	
2060: A0					BF	00	204 4E >		2340:40		BE	02	A9	80	BD		
2068: F7	CØ	C9	4A	FØ	68	20	EF<90>		2348:02	20	1E	21	CØ	90	30	98<63>	
2070: E4						4C	90k BF >		2350:A4		DØ	ØE	C9	41		2B(CØ>	
2079:20				7F	7F	-	CF (35)				98	E3	09	39	BØ	DEKE9>	
2080:03				CE		BE	00KA42		2360:90		C9	7E	00	0D		18<28>	
							4C: B6>									A5 <b8></b8>	
2088: 20				44				i i					16	C2			
2090:96				85		A9			2370:40				9B			C9<67>	
2078:65	AC	AZ	88	81	A8	29	@7 <ab></ab>		2378:30	90	C4	C9	3A	90	84	C9 <e1></e1>	
20A0:09				1.6	A9	AØ	03<94>		2380:41	98	BC	€9	58	80	88	CØ<28>	
20A8: C9							ØD <ba></ba>	1	2388:08			99				A9<6C>	
2090:F0				FD	ØA.		C9 <c5 <="" td=""><td></td><td></td><td></td><td></td><td>Ć2</td><td></td><td>20</td><td></td><td>C2&lt;1B&gt;</td><td></td></c5>					Ć2		20		C2<1B>	
2008:08					DØ	-	88<1F>		2390:E6							83(21)	
2000:18							<b>02</b> <03>				43	99	84	04		54 <d6></d6>	
20CB: E6	AC	C5	80	A5	AC	E5	81(39)		2348: 99	85		A9	42		-	04 <e6></e6>	
2000190	C9	4C	A3	F6	B1	AB	C9< A0>		2300: A9	9B	99	87	04	A9	44	80<96>	
2008:07							C9<68>		2388:80			A7		81		A94E6>	
2008:07							C9< 2E >		23C0:3A		82		49			54(80)	
														-			
20F8:27							C9(84)		2308:03			90	55	82		08<53>	
20F0:3F				2F	DØ		4C+8Ø>		2300:0D		23		22	GD.		03<5A>	
		AC	A6	F5	4C	85	F6(92)		23081A9	23	20	BF	F9	AØ.	FB	20 <f7></f7>	
20F8: A3	FS								23E8:E5		0.00	an.					
20FB: A3 2100: 40				84	20	A2	82<4E		TORBLED	1 13	HMO	40	ZW	E5	FB	A4KCE>	
2100:40	36	E5	A2														
2100:40 2108:49	36 A9	E5 91	A2 AB	18	CB	B1	AB<91>		2350:00	20	E5	FB	Α4	81	20	E'5< DE >	
2100:40 2108:A9 2110:75	36 A9 00	55 91 91	AB AB	16 40	<b>63</b>	B1 B1	AB<91> AB<88>		2350:60 2360:68	20 A4	E5 86	F8 20	A4 E5	81 F8	20 A4	E2 <de></de>	
2100:40 2108:49	36 A9 00	55 91 91	AB AB	16 40	<b>63</b>	B1 B1	AB<91>		2368:80 2360:68 2368:20	20 A4 E5	E5 86 F8	FB 20 A4	A4 E5 88	81 F8 20	20 A4 E5	E5 <de> B7<a3> F8&lt;47&gt;</a3></de>	
2100:40 2108:A9 2110:75	36 A9 00 01	E5 91 91 91	AB AB AB	18 AØ AØ	CB 83 84	B1 B1 4C	AB<91> AB<80> 1F<5E>		2350:60 2360:68	20 A4 E5	E5 86 F8	FB 20 A4	A4 E5 88	81 F8 20	20 A4 E5	E5 <de> B7<a3> F8&lt;47&gt;</a3></de>	

2408:54 03 A9 22 8D 55 03 38(6E) 2410:A5 88 E9 6F BD 58 03 A5(09) 2418:89 E9 22 80 59 03 A9 0B(OF) 2420:20 BF F8 20 BD FB 20 EF<2A> 2428:E4 9B 20 20 20 20 4E<16> 20 2430:6F 63 68 20 65 69 6E 6D<65> 2438:61 6C 20 2440:68 65 72 73 78 65 69 ASCRAS 2B 4A 2F<4C> 6E 20 29 20 3F 80 20 1E 5F C9 4A F8 85 C9 2448: 4£ 21(97) 2450:29 4F (07) 2450:DØ F3 68 40 47 A9 @C(83> F7 2460: A2 10 80 52 03 20 AF 20(8A) 2468:30 01 60 98 48 20 EF E4<1B) 6F(58) 2470:49 2F 4F 20 45 72 72 2478:72 20 2D 29 98 68 AB 88 4A> 2490:20 E8 C1 4C 47 F7 A2 10(11) 2488:20 FE 20 C0 00 30 DC 60(38) 2470:A9 04 A2 88 20 FF C3 A9(16) 2498:FF 91 AB CB 91 AB A5 80(6E) 24A0:CB 91 AB AS B1 CB 91 ABKCC> 74A8:A9 00 20 84 E9 A9 9B 20(04) 2480:A9 E9 20 F1 C4 A2 B2 A9(EF) 2489:03 20 FF C3 A9 FF 91 AB(A4) 2400:08 91 AB CB A9 00 91 ABK7B> ED< 75> 24C8: 60 00 DE FF DF ED E9 24D0:F7 OF FA BD 20 22 22 DD 0E> 2408: 21 AA EA C1 EA C1 FA 42(EC> 2460:F0 EA EE E8 EE CS EE 97(F8) 24E8:EC 84 EC 87 EF EF ED 28<41>
24F8:EF EB ED 57 21 68 21 C6<38> 24F8: ED ED DE ED DO ED 2500:ED 7D ED 54 C9 A9 FA 98 88> 2588: FA AZ ED 66 FA 88 F9 84<6B> 2519:ED 78 ED C1 CA 48 EF 48(FE) 2518:EF CE C7 20 F7 32 F8 28(91) 2520:F8 ED F8 F4 ED 2A EE 06(E1) 2528:EE LE EE FF ED BC EE B1<C1> 2530: EE &F F1 96 F1 2E F1 Ø6<20> 2538:F1 69 F1 96 F1 ZE ED 40(22) 2540:ED 47 ED 86 EA 38 F6 89(89) 2548:FR BE FA 07 EF 10 EF 33<B2> 2550: EF 36 EF 30 EF 39 EF 95(43) 2558:FB C1 EF C3 EF 4A EE 4E(24) 2560: EE 78 EE 12 EE 50 FC 53.9A> 98 FC 9B FC 50 FC 53(37) 2568: FC 2570:FC 92 FC 95 FC DF FA 86(BB) 2578: FØ B1 EC AB EC E4 FA 00.02> 88 88 88 88 2580:00 00 88 OOK CAD 2588:00 00 00 00 00 00 00 00 00(D2) 2590:00 00 00 00 00 00 00 60(DA) 2598:00 00 00 88 88 00 00<E2> 25A0:00 20 00 00 00 00 00 00 00<FA> 25A8:00 00 00 00 00 00 00 00 00(F2) 2590:00 00 00 89 99 28 ØØ (FA) 00 2588:00 00 00 00 00 00 00 00 00<02> 2500:00 88 88 88 88 88 80 80 80 80 2508:00 00 00 00 00 00 09 F1 C3<DA> 3B(D7) 2500:F1 BB F1 CF F1 BS EB 25D8:EC 77 EB 96 EB 83 EC 94KEE> 25E0:FE 55 FF D2(B7) DA FE SE FF 23E8: EA 68 FF 51 EB 74 FE 17<F2> 25F0: FE 86 FE 33 FE 92 FE AAK RD> 25F8:EB 10 FE 2C FE 48 FE 4E<99> 24001FE 06 FD 98 EC 18 FD 6AKE9) 2608:F2 76 F2 96 F2 90 F2 BAK7E) 2610:F2 A0 F2 38 F5 0E F2 FC(2F) 2618:F1 85 F2 17 F2 8D F2 B5(E9) 26201F2 AA F2 F5 EA 13<D4> CE ED 2628:00 3E EB D5 F2 DE F2 F5<7E> 2638: F2 E5 F2 EC F2 FC F2 82(11) ACKA4> 2638:FB 57 F2 54 F6 82 F4 2640:F4 6F F6 D9 EB 55 FD 21(86) 94(81) 2548:F4 4A F4 89 EC 1A EC 2650:F3 09 E9 DD C3 0C F5 06<792 2658:FS 31 21 00 F5 5E F4 75(09) 2660:F3 1E F5 12 F5 18 F5 24<83> 2668:F5 C7 FZ 68 F2 C1 F4 D6(89) 2670:F3 DA F3 DE F3 E2 F3 40 ( 2F > 2678:21 08 F6 14 F6 72 F6 34KEC> 2680: F4 28 F6 16 F4 A4 F3 A9<AC> 2688: F4 BE F4 25 FB 67 F2 73<0B> 2690:FZ 93 FZ 7D F2 87 F2 2698:F2 38 F5 WB F2 F9 F1 @2(92) 26A0: F2 14 F2 96 FF 9D FF A4KRE> 26A8:FF AF FF E5 F1 D5 F1 DB<F6> 26B0:F1 IB C7 34 F2 E7 E1<4E> 2688:F1 22 F2 03 EC 20 EC 46<DD> 2ACO:F2 NO NO NO DO DO DO DO NO(D5> 26CB: 00 00 00 00 00 00 00 00 00(14> Laenge 9933 Bytes

Listing 1. Turbo-Basic-Compiler (Schluß)



#### Beginn: Listing 2 Turbo-Basic-Compiler-Runtime-Paket (Länge: 10876 Byte)

02C8:00 46 DD F0 0E 81 DA 91<8A> 0000:FF FF 7D 20 C5 20 9C 84(BF) 20 02D0:06 E D0 F9 E6 D7 E6 DB<AC> 0008:00 A9 7D AØ. 63 0010:00 A9 FE A6 DC FO 88 BIKEE> BD 81 D3 20L DAV 02 > 9208:CA DE F2 **C8** CA F8 A9(A3) 02 6C(C2) 02E 0: DA 91 06 90 E2 0018:EE 4C ED F9 6C 81 D3 20 93 20: BA 02E8:FF 80 61 **D**3 30 A2 10(D1) 02 0020:E0 EE E4KEC> DC 96(83) 02F0:A9 90 42 D.T. 4C 54 0028:4C 22 21 EE 01 20 OC. 4C F2 0030:20 22 21 EE 01 03 20KE2> MOTER: AC 200 (2)(2) 82 EB 02 BR(E2) 84 A9 85 85KF92 0008:56 E4 4C 22 21 20 AA EA A62 0380:60 98 CØ CP 84 0308:F4 01 03 6C 90 28 AA: D3 200 80 85 F3 60 Δ4 F2(BD) 0040:EE C9 60(59) EE 01 03 4C 71 **E4** 00KEF 0310:61 F3 38 E9 30 ØA. 0048: EA 20 22 21 4C<BE> Ø318:84 F7 AB 26 89 E9 FB 98(88) 0050:21 10 21 £A FB<0C> 2A 28 0320:F0 2E 4A 48 44 4A 65 005B: B1 EØ. 44 SA 20 9B<AA> 98 SA 9B 0328:98 0.7 AØ 20 98 62 69<1A> 3A 50 45 3A<61 × 0060:53 0330:07 46 F8 90 02 69 15 46(24) 006B: 9B 80 EE Øl D3 6C D4 00<73 90(07) 21 71 29 FC 22 98 48 AD R14550 0338:FB 100 62 69 31 46 F8 0070:22 CB 46 FB<36> 09 02 8D D3<57> 0340:05 69 63 90 01 007B: D3 21 D44 BB / 27 01 28 60 01 D3 0348:90 200 100 CB 90 08(61) EE 85 0030:69 0350:A6 FØ 6F FŻ 98 07(10) 2088:84 05 AB 22 D4 CE 01(85) F7 46 0090:D3 60 EE 01 D3 85 D4 84×FC2 8358:69 200 CB CB 98 81 CB 46(C3) Ø360:F7 AA 98 69(26) 90 98 69 AØ Ø1 B1 D4 A8 AZ 11 2099: DS 20 DA 01 60 00A0:A1 D4 03 40 29<00> 0348:05 A8 BA 46 98 68 69(D3) CE 0370:23 00 D0 F9 D3<60> 81 1000 98 69 10 AR RA 46(39) 00A8:F5 CØ EE 98 69 47 AA 98 69(E0) MORO: 91 88 AD 81 D3 29 EC. 89<41> 0378:F7 98 D3(B9) 0380:20 AB SA 46 90 88 69(A5) 80 01 D3 60 EE 21 00H8:02 0788:95 40 A8 8A 91 AA 98 100 98 69 46<69> 80C8: A2 00 81 68 98 40 01 DØ4 3F 2 68 00 CE 01 DE AR AD- RES 0.390±F7 90 P.C 49 69(11) 98CB: F7 84 90 02 0398:01 E6 D5 46(2A) D4 10 -0000:20 ØF. 0.3 6C 80 62 F1> 00D9:48 BA 48 A9 21 40 A9 77(C1) 03A0:F7 90 0E 69 83 AA 98 694B1> 00E0:48 BA BD 05 01 48 DB 49, 305 DTAR: A3 AB A5 05 69 DI S 85 05(79) 0.5E0:8A 46 BE 67 AAKB2> DOLD: SA 48 98 48 FE (7) S D.S. BD/DD> E7 92 69 100 27 00F0:0F 04 22 02 48 49 0360:98 AB A5 05 69 03(92) 6C 21(C2) 84 D6 79 A9 48 88 EE 01 D3<47> 03C0: III D5 88 95 D7 A9(FF) 00FE: 48 16 02 EL 01 CE 01 DC 60 18 0120:6C 20 20 REs 2R> **0.5CB: 4**III 0.4 4C FE Cl 60K 3A) 85 47 03D0:A5 49 80 ED A5 EB<17> B108:21 BD 23 E2: EØ 40 03<25> FØ F2 85 E7 A5 04<575 0110:48 BD 46 03 48 BD **6**3D6: 29 7F 2A BD 48 03 BS 0116:85 **7B** 98(00) **MREA: 29** 7F 38 E5 E7 RØ 33 ASCR6 > 03E8: D4 84 D4 A5<83> 28<C1> 85 EØ 0120:AD 06 A4 EØ A4 05 00 00 CE 01 03 60 85< 80 > 03F 8: 05 E I 25 E1 84 A5KZ67 0128:20 01 30:00 84 8D EE 01 03 A6 80×F5> 03FB: 06 04 E2 85 E2 84 D6 A5< 050 0400: D7 AS(FD) DIESELA BA AS 89 85 89 1.8 BACZE> **A4** E3 85 E3 84 97 84 E4 E4 85 88 E8 94 0408: DB De A5<000 @140:65 BB 0C FØSAB2 91 0148:25 89 B1 88 AB 98 DØ(49) 0418:09 A4 E5 85 E5 84 D9 4C<C9> 88 FB FB 4B DISO: ER ER 1 B C6 89 C6 RB RR (RA) Ø418:D1 CØ AB 36(3B) 0420:88 FØ @158:B1 88 91 8A 88 B1 88 91<FB: FØ 22 88 1.0 68 D0<C5> 6478: AS A5 E1 E5 0160:8A 88 B1 88 91 BA 88 B1<29 85 84 B4<D9> DØ492 Ø168;88 91 BA 99 DØ E9 CA 0430:EX 84 E2 4C 61 CI A5 E2<C4> 8D<87> 0170:E2 4C 22 21 85 BC 84 0438:85 E5 A5 E1 E4 85 84 E3<4E5 FR(A7) E2 4C Ø178: EE 101 D3 AB 00 A6 BD 8440:84 61 C1 A5 E3 85×F93 0180:10 B1 88 Q1 BA CR B1 BS(CC) 0448:E5 AS EZ 85 E4 A5 E 1 85 (B1) 91 0188:91 8A C9 BI 88 84 C8<5D> 0450±E3 84 E2 40 61 C1 A5 E4<27> E6(57) 0190:81 88 91 BA Ca 00 EA 0458:85 E5 A5 £3 85 **A5** E4 E2<8F> CA A198:89 88 DØ FZ A6 BCKAR > 0460:85 A5 E1 85 €2 84 E1<890 MIA0:F0 08 B1 88 91 BA CB CA<47> 0468:FB A5 D4 45 E0 365 30 18KAC> 0148:00 FB 4C 22 21 EE 01 D3<D1> 0470:A5 D9 65 E5 B5 D9 A5 DE AB> 0190:20 E6 DB CE 01 03 60 EEK 58: 0478:65 E4 85 A5 D7 £349@> DB 65 0188:01 D3 20 00 08 CE 01 03(BE) 32 61(6A) 0480:85 D7 A5 D6 65 85 D6<A4> E2 60 0100:50 20 ED E9 88 8488: A5 05 65 E1 85 05 98 16(E9) 0108:AE 24 E4 25 E 4 E8 DØ< 3A> 0490: AS 85 DB 85 D9 **A5** D7 DB(13) 21<90> 0100:01 CB BE D7 21 80 DB 0498:A5 D6 85 D7 A5 D5 65 06<62> D4 4C E5 85 Ø108: A9 99 BD 8E **D4** 78 49 FE(73) 04A0: A9 91 85 D5 E6 EC. 9E> @1E0:80 @1 03 A9 7Ę BD FA FF (6F) 04AB:C1 38 A5 D9 D9(E2) E5 21 8D FB FF A9 A3 80(19) 04R0:A5 D8 E5 01E8: 49 E4 85 DB A5 D7<A3> MARB: E5 DIEG-FE EE A9 21 80 FF FF 49. FB E3 85 D7 A5 D6 E5 E243A> 01F8:CC 20 98 60 A9 EØ 88<8D> 20 04C0:85 D6 A5 D5 E5 E1 85 D5<08> 0200:60 A9 40 BD 0E D-4 58 A94.383 04CB: BØ 27 A5 D4 49 88 D4<6B> 85 0208:FF 0408:38 ES(AZ) 8D 01 D3 20 AE 60 49(5F) 98 E5 D9 85 ng 78 44 80<7A> ØD 04D8:D8 85 D8 98 E5 D7 04F0:98 E5 D6 85 D6 98 0210:00 B5 09 95 OC. 85 D7< 05> 0218: 7E 20 80 7F 20 A9 70 A0<28> 99 E5 D5(A6) FE 04EB: 85 84 49 BDK B3> 0220:20 85 ØC. ØD D5 4C EC C1 86 DA D8<D92 04F0:02 04F0:A5 0228:01 D3 A2 01 BE FB 93 RAC9E) AZ 88 DB A4 D4 F8 6D(A9) 44 79 A2<7A> 05 00 A5 D6 CA 92 69 SC 0230:09 BE 88 DØ<16> F4 @238:60 AB ØF 20 53 40 86<C6> 0500:47 A5 D7 DØ 260 68 A5(86) 55 54 D9 DØ 0240:F7 44 31 3A 41 54 AFKE20 0508:08 DØ 18 88 A5 188 (BE) 9BKBA> 43 42 Ø248:52 55 4E 2E 0510:BB FØ 51 85 D5 D0 40k 3E 2 06 0250:85 05 AB 00 84 D4 A2 84<92> 0518:85 86 A9 80 09(57) 05 85 0258:A9 FF 8D 01 D3 81 04 99(80) 0520:85 DB 85 D7 FD 52 85 D5<273 D3<87> 9249:99 95 CB DØ F8 CE 0.1 0578: A5 1000 85 D6 86 D.Z A9 ROKERS. D4 85 91 F8(C9) PIP CB DØ Ø268: 89 0500:85 D9 DB FB 85 22 85 D5<27> E3 10-BE> 0578:AS A5 09 85 07<09> 0270:E6 D5 CA DØ 60 AZ 100 65 06 0278:49 DA 9D 44 613 49 00 90(95) 0548: BA DE AS 00 RS D9 FØ 10 BB) 49 9D A9 9D(83) 0548:85 0280:45 03 0.3 04 100 A5 D7 85 D6 A5 DB ( BC ) 0550:85 D7 0288:48 03 99 07 9D 42 63 28<86> **A5** D9 85 08 86 D9< E83 0290:56 E4 301 50 49 33 85 DACABO 0558:84 D4 99 29 7F CS 71 B8<46> 44 03 49 DB 90KCF> 0298:9D 61 85 0560:08 C9 ØF BE **B**3 29 29 E9(21) 02A0:45 03 38 **A5** DE E.3 D6 85< £8> 0569:18 86 FC 84 FD 85 EF<De: 60 85 9578:28 02A8:DC A5 D9 E5 D7 C9 C2 29 3F EC EF(D4) DD E6<98> **C6** 0/280: DC D0 02 E6 DD A5 DC 90(97) 0579:29 BB E4 BB 37 A5 FC 69(8A) 23 02B8:48 03 A5 DO 90 49 20(54) 0580:06 85 FC 98 82 E6 FD 28(29) 01 D3 A0< DA> 0588: TF EC 20 D1 C0 80

0590:EF F0 21 AD E0 05 85 E0<00> 0598:AD E1 05 85 E1 AD E2 05<20>
05A0:85 E2 AD E3 05 85 E3 AD<70> AD 05AB: E4 25 85 €4 E5 05 85<5C> 05B0: F5 4C 60 A5 D4 8544F) OSBB: FO AS 05 85 E1 A5 DA 85(99) E3 A5 85<2E> R500: E2 A5 DB D7 85 05C8:E4 **A5** 09 85 E5 60 A5 D4<59> EØ 05<A3> 05D0:8D 05 A5 05 80 EL 05DB: 05 D6 BD E2 DS 45 D7 ADCAE) 05E0:E3 05 AS D8 8D E4 05 A5<66> A5 @5E8:09 8D €5 D4 8D<51> 05 60 05F0:E6 25 A5 05 80 E7 85 AS(CB) 05FB: D6 8D F8 05 45 07 BD F9(A2) A5 D9(51) 0600:05 AS DB BD EA 05 0698:0D EB 25 60 A5 EB BD E6KE1> BD E7 05 E3 8D 0610:05 A5 EI A5 E2<31> RA18:80 EB 85 45 E9 05(FF) E4 0628: A5 BD EA 85 A5 ES 80<30> 0628: EB **8**5 A2 90 FO 05 60 96:0549 20 A2 ØC. BD EO RS RS(13) BD E2<A4> 8D E1 05 0A 5B: 04 85 DS E3 85 D7<C1> 0640:05 85 D6 BD 85 8648: BD E4 05 85 DB BD 23 05(15) 0650:85 D9 60 A2 @C 2C 42 DA(45) BD 63 El 0658: BD E Ø Ø5<9C> 205 85 E2 8660:85 Εı BD 05 65 E2 BD<78> Ø668:E3 05 85 E3 BD E4 25 BS(BA) A2KAD> 06/0:E4 BID E5 05 85 ES 60 0678:05 28 **C4** 95 EØ CA 10(E9) BD 20 08 E4 90 F0 F1 A5 D4 85 F0 0680:F8 49 99×C6> MARR: 85 29 7F < 1A> DØ 0690:85 D4 DEK DD> C9 40 90 16 29 0698:A5 05 FØ 40 85 Fi 4A(61) 0660:46 65 Εļ BS. F1 05 05 29(82) F1 85<7A> F1 DAAR: DF 65 85 A9 20 EF<62> @6F0:05 20 EC C1 A9 ØA. A2 0688:A0 C 2 98 E44223 20 65 C2 20 00C0: A5 F1 FR OF 44 18 45 D4<9B> F1 0345A> 8008:30 85 04 46 90 16 67 C7 0600: 20 86 FB 90 20< D1> 0608:80 C2 20 DD C3 40 6D £5(16) 01(95) 86 87:6369 49 40 85 D4 AB. **86EB: 84** 05 84 84 07 88 06 60 Ø6F Ø: DB 84 3D 94 19(71) D9 17 96 69:B349 310 57 33 285 98 200<BE> 76 32(11) 6700: 3E 05 54 62 3E 66 27 80 0708:19 62 36 91 88 68(2C) 0710:30 36 3F 07 32 **B**3 27 41(RE) 43 3F 0718:3F 25 34 56 75 66(27) 37 12<DC> 0720:27 50 50 40 21 15 0728:92 55 3F 99 99 99 99 994F75 0730: SE 43 42 94 48 19 86 FC(DF) FD ECKD0> 0738:84 20 C9 02 20 3F 0740:20 D1 CB 20 E.B 20 26<39> 0748:C3 2**0** 3E EC 20 CB CØ 20<13> 4C 0750:51 6D 65 28 60 A9(27) 0758:05 DØ 02 A9 00 85 F Ø A5(3E) 0760:D4 30 0768:85 F1 F 2 FØ. FØ ØA 40 88K5B> 89 A5 D5<9F3 40 85 D4 0770:29 FB FØ 95 E6 20 6F < 81 > Ø7781C7 A2 02 AØ C4 20 31 C4KCA> 2780:28 FB C2 20 MR FA 69 **MAKEED** 0788:A2 DB C2 28454> AB **C4** 65 20 0770:51 C3 20 0B E4 A9 3F 85<86> 85(41) 0798:E0 A9 50 85 E 1 49 DED 07A0:E2 B5 F 3 85 E4 85 E.5 28(04) 0748:D1 C8 20 CZ A5 F1 10(B4) BØ 0780:05 10 49 EE 69 01 68 **OPKEES** 0788: 20 1.73 0.01 A5 F1 29 67 R5<20.5 07C0:04 85 28 D4 D1 CØ 86 F0K18> 07C8:F0 BD ØB **2B** C4 EØ CAKD6> 4C 0700:10 FB 6D E5 18 68 465 RD > 15(7F) 87D8:93 16 22 77 66 3F 49 Q7E 0:57 **8**B BF 51 70 11 39 0718:08 3F 20 57 95 RECDES 61 39 89(3C) 07F0:04 63 0.3 55 3F 10 07F8:30 35 09 39 04<50> 12 64 ØB. QBQQ:60 3F 12 42 3F<11> 58 47 42 0808:17 37 17 849 08 3E 28 95<365 71 88 96<8A> 0810:29 17 3F 86 85 0818:44 3E BEKEES 16 05 44 49 00 0820±95 68 38 45 00 3F 02 68(89) 0828:79 94 BF 104 92 78 90(73) 16 8859:88 3F 07 23 15 20 00 BFKC90 0858: 2B 92 29 12 44 3F 11 RB< 783 0840:40 09 11 14 28 31 BF 56< BA> 0848:04 3F 19 99 98 44 BF (B4) 0850:33 33 33 31 99 1.5 3F 99(57)

Listing 2. Turbo-Basic-Compiler-Runtime-Paket. Geben Sie das Programm bitte mit AMPEL von Seite 87 ein.

25

C6KIF)

02C0:56 E4 30 34 CE

0000-00 00 00 TE 30 E3 00 17:00		
Ø858:99 99 99 3F 78 53 98 16<28>	0830:C7 08 80 06 65 8A C5 E0<8F>	ØEØ8:48 98 CF F6 D9 C9 82 F8<23>
0860:34 38 60 A9 04 24 04 10<06>	0838:90 CA A5 88 69 07 85 88(AB)	0E10:45 BØ 42 A5 D6 29 FØ 4AKFA>
0868:06 A9 02 D0 02 A9 01 85<1C>	0840:98 82 E6 89 E6 D4 D8 82-84>	0E18:85 D4 4A 4A 65 D4 85 D4(33)
0870:F0 A5 D4 29 7F 85 D4 A6(A2)	0848:E6 D5 A5 D6 D8 B2 C6 D7(89)	0E20:A5 D6 29 0F A6 D7 E0 50(BA)
0878:FR 8D 14 C6 85 E0 8D 15(8A)	8E50:C6 D6 4C D2 C7 40 28 68(58)	
		9E.28:65 D4 85 D4 98 29 BF AA(0B)
Ø880:C6 85 E1 80 16 C6 85 E24C1>	ØB58:00 A8 0A 0A 2A 2A 29 03<85>	0E30:BD A1 CB 65 D4 85 D4 BD <bb></bb>
0888:80 17 C6 85 E3 80 18 C6(4E)	2860:AA 98 50 50 C8 60 18 18(18)	0E38:AB C8 65 D5 85 D5 98 29(85)
0890:85 E4 8D 19 C6 85 E5 20\AF	8868: 9C 18 30 30 60 68 C0 C8<07>	BE48:FB FB 12 4A 4A 4A AA AS ASCA4>
0898: 6D E5 B0 C5 A5 04 29 7F(24)	0870:00 C0 18 0C C2 C0 28 14 <ff></ff>	9E48:D4 7D 8D CB 85 D4 A5 D5(A1)
08A0:38 E9 40 30 24 C9 04 10 5E>	0878:14 8A 8A 14 14 28 28 28(57)	0E50:70 9E CB 85 D5 60 C0 07<6E>
08A8:88 AA 85 05 85 F1 29 10475>	0880:28 28 28 28 14 28 80 08<70>	0E58:80 FB A5 D7 29 F0 4A 85<4F>
0890:F0 02 A9 02 18 65 FL 29(CB)	8898:00 02 03 02 03 02 03 01 <c6></c6>	0E60: D4 4A 4A 65 D4 85 D4 A5<7D>
0898:03 65 F0 85 F0 86 F1 A9<1E>	0890:01 01 00 00 03 02 00 01<66>	8E68: D7 29 8F A6 D8 E8 58 65(3A)
0000:00 95 05 CA 10 FB 20 EC<14>	0898:03 07 00 F0 FC FE 04 02(D2)	BE78: D4 B5 D4 B9 B1 CB 65 D4 <fa></fa>
0808:C1 46 F0 90 09 20 80 CZ(6D)	09A0:01 38 50 A5 57 29 0F AA<92>	@E78:85 D4 89 88 CB 65 D5 85(86)
0800:20 DD C3 20 CB C0 20 E0:02		
	0848:49 80 85 DF A5 BA 84 26(1E)	0E80: D5 A4 D6 D0 A7 60 00 10 <ae></ae>
0808:02 20 08 E4 B0 83 A9 864F92	@BR@:DF @A 26 DF 65 8A 98 @2 <df></df>	BEBB: 20 38 40 50 68 98 27 4E<24>
08E0:A2 F6 A0 C5 20 65 C2 20(03)	0888:E6 DF 0A 85 DE 26 DF 8D<45>	0E90:75 9C C3 EA EB 03 00 07(37)
08E8:51 C3 20 0B E4 46 F0 90(2E)	GRCG: 61 CB 65 E8 C5 GA F8 D9(CE)	0698:88 88 A8 8F 88 13 70 17 <c0></c0>
08F0:09 18 AS D4 F0 04 49 80(1A)	88C8198 D7 BC S1 C8 84 E2 89<86>	0EA0:58 18 48 1F 28 23 00 64<08>
08F8:85 04 60 BD 03 55 14 99/E2>	ØRDØ: 95 C8 8D C0 65 RD 71 C8<98>	DEAB: CB 2C 90 F4 SB BC 20 B4<5A>
0900:39 3£ 01 60 44 27 52 BE < 75>	8BD8:85 E1 4A 4A 4A 4A AA F8(8E)	0EB0:00 00 00 01 01 01 02 02(EE)
0908:46 81 75 43 33 3F 07 96(56)	0888:07 06 DE 26 DF CA D0 F9 <d2></d2>	0698:03 83 80 E4 BD FF 18 4C(B3)
0910:92 62 39 8F 64 59 64 08(26)	@BE8:10 A5 DE 65 58 85 DE A5<27>	0FC0:09 E9 18 68 38 68 20 B0 <c6></c6>
2918:67 40 01 57 07 96 32 40-A6>	@BF@:DF 65 59 85 DF A5 89 85(A4)	BECB: C2 A5 D4 FB F5 A5 EB FB(96>
0920:90 00 00 00 00 3F 01 74 ED>	09F81E4 A5 88 85 E3 A4 E2 84(B0)	DED0:ED 45 D4 29 80 85 EE A5(4C)
0928:53 29 25 A9 00 85 F0 85 E8>	0C/80:ED F8 9C 39 91 C8 85 ED<3D>	0ED8:E0 29 7F 85 E0 A5 D4 29 <c9></c9>
0930:F1 AS D4 29 7F C9 40 30000>	BC88:46 E4 66 E3 88 DB F9 A5<69>	0EE0:7F 38 E9 40 38 65 E0 30<72>
0938:15 A5 D4 29 80 85 F0 E6427>	0C10:E4 D0 9E A5 E3 C5 E1 B0 <c6></c6>	WEE8: DB W5 EE AB 28 08 E8 85(FD)
0940:F1 A9 7F Z5 D4 85 D4 AZ<53>	9C18:88 A6 E2 D8 BA A5 C8 28(A9)	0EF0: DA 85 08 85 DC 85 DD 85(81)
0948:50 AD C5 20 31 C4 20 E8(85)	0C20:54 C8 8D C8 65 1B 60 BC<67>	
		@EF8: DE 85 DF 84 D4 A8 07 46<01>
0950:C2 20 08 E4 80 38 A9 08451>	0C28:91 C8 84 E4 80 98 C8 85(CA)	0F00:E5 90 30 18 A5 DE 79 EE(B9>
0958:A2 14 A0 C5 20 65 C2 88<2A>	0C30:EE A5 CB 0D C0 05 40 C0<9E>	0F 08: 20 85 DE A5 DD 79 E6 20(26)
0760:30 20 51 C3 20 00 E4 80 00>	8C38: 85 99 C8 85 A6 EE 8A CA(4D)	0F 10:85 DD A5 DC 79 DE 20 85(A1)
0968:28 A5 F1 F8 13 A2 05 BD<7E>		
	0049:00 FC 88 10 F4 A4 E4 AD(18)	8F18:DC A5 DB 79 D6 28 85 DB(48)
0970:56 C5 95 E0 CA 10 F8 20(1F)	9C48: C0 05 99 C0 05 A6 EE 38<97>	0F20:A5 DA 79 CE 20 85 DA A5<84>
9978:D1 C8 A5 F8 85 D4 85 D4(6D)	0C58:2A CA D8 FB 8B 10 F3 16<1C>	0F28:D9 79 C6 20 85 D9 88 18<3A>
0780:A5 FB F0 0D A2 05 BD 20(AF)	@C58:68 85 8A 98 D0 FA 20 9E(68)	0F30:CE 30 05 F0 03 88 10 C7(C1)
0988:C6 95 E0 CA 10 F6 20 6D <e9></e9>		
	9C60:C8 B0 F5 A5 0E 85 8C A5<73>	@F38:A@ @7 46 E4 90 30 18 A5<@D>
0990:E5 60 38 60 18 60 A9 00 <cb></cb>	@C68:0F 85 80 AD E5 02 E9 06 <ec></ec>	8F40:00 79 EE 26 85 00 A5 DC(81>
0998:85 F1 AS D4 30 F4 F0 F4 <c0></c0>	0C70:85 E7 AD E6 02 E9 00 85<51>	0F 48: 79 E6 20 65 DC A5 DB 79<00>
09A0:C9 3F F0 05 18 69 01 85<22>	0C78: E8 18 A5 8C 67 03 85 8C <dc></dc>	0F50: DE 20 85 DB A5 DA 79 D6<19>
09A8:F1 A9 06 85 EF A9 3F 85<83>	0C80:98 62 E6 8D C5 E7 A5 8D<9F>	0F58: 28 85 DA A5 D9 79 CE 28(61)
0990:04 20 E8 C2 20 B0 C2 20<41>	9C88: E5 E8 90 03 4C 61 FA A6 <c9></c9>	0F60:85 D9 A5 D8 79 C6 20 85 <f1></f1>
0988:DD C3 E6 05 20 C8 C0 20(C1)	0C90:ED A4 E3 20 AB CA F0 03(AD)	0F48:D8 88 18 CE 38 05 F0 03<2C>
09C0:51 C3 20 0B E4 A5 04 80 <cb></cb>	9C98:4C 3C CA 20 9D CA 20 96<18>	0F70:88 10 C7 A8 07 46 E3 90<50>
09C8:EC 05 A5 D5 BD ED 05 A5 <bb></bb>	8CA8: CA 98 30 00 20 AB CA D6<3D>	9F79: 38 18 AS DC 79 EE 20 85<47>
2900:06 80 EE 05 A5 D7 80 EF (CB)	BCAB:86 29 9D CA 4C 99 C9 28(92)	0F80:DC A5 DB 79 E6 20 85 DB<7E>
0908:05 A5 DB BD F0 05 A5 09<20>	0C98:8D CA 98 A8 BB 91 8C 8A(83)	@F88: A5 DA 79 DE 20 85 DA A5<6A>
09E0:8D F1 05 20 B0 C2 20 2A<0E>	BC88: 0A 0A 0A CB 91 BC A4 E3(F3)	0F90:D9 79 D6 20 85 D9 A5 D8 <df></df>
09E8:C3 20 AD E5 20 4E C3 20<40>	0CC0:A6 ED 20 9D CA C4 E1 B0<82>	0F98:79 CE 20 85 D8 A5 D7 79(AB)
09F0:CB C0 20 18 C7 AS D4 F0KEF>	BCC8:08 20 AS CA D0 06 20 90<7B>	0FA0:C6 28 85 D7 88 10 CE 30(11)
0988:00 20 4F C3 20 D1 C0 C6(89)	0CD0:CA 4C 8D C9 20 96 CA 98(AB)	0FA8:85 F0 93 88 10 C7 A0 07<10>
0A00:EF 10 C2 30 03 20 20 C3<80>	9CD8: A8 82 91 8C 8A 88 11 8C(58)	0F80:46 E2 90 30 18 A5 D8 79(FD)
0A08: A5 F1 F0 OF 4A 18 45 D4(6B)	9CE8:91 8C A4 8A C8 C4 E9 B9<56>	0FB8:EE 20 85 DB A5 DA 79 E6(79)
0A10:E9 1F 85 D4 46 F1 90 03<70>	0CE8:23 20 60 CA 20 7C CA A0<44>	0FC8:28 85 DA A5 D9 79 DE 20<1F>
0A18:20 67 C7 18 60 4C 09 E9(4B)	9CF0:01 B1 8C 29 97 C5 ED C8 <e6></e6>	0FC8:85 D9 A5 D8 79 D6 20 85(59)
@A20:A5 D4 29 7F C9 @F 9@ F5<57>	MCF8:81 BC 65 63 90 0B A0 00<7E>	0FD0:D8 A5 D7 79 CE 20 85 D7<55>
0A28: 20 50 C2 FB A2 86 A0 04(88)	8D90:B1 8C 89 88 91 8C 4E 74 <bc></bc>	@FD8: A5 D6 79 C6 20 85 D6 88(B1)
0A38:18 A5 D9 65 E5 85 D9 A5 88>	8098:C9 20 6E CA A4 8A 88 C4<19>	0FE0:10 CE 30 05 F0 03 88 10<21>
2A38: D8 65 E4 85 D8 AS D7 65<7E>	@D18:E0 B0 20 20 6E CA 20 7C(62)	0FE8:C7 A8 87 46 E1 98 38 18<04>
0A40:E3 85 07 A5 D6 65 E2 85(E5)	9D18: CA A0 01 01 0C 29 97 C5<46>	0FF8:A5 DA 79 EE 20 B5 DA A5 <b1></b1>
0A48:06 A5 05 65 E1 85 05 8A(E2)	8028:ED C8 81 8C E5 E3 98 88(60)	0FF8: 09 79 E6 20 85 09 A5 08(68)
0AS0: 69 00 AA 88 D0 DB D8 8A<6B>	0028:A0 00 B1 8C 29 7F 10 04<18>	1000:79 DE 20 85 DB A5 D7 79<20>
9A58:F8 1A A5 D8 85 D9 A5 D7 <c6></c6>		
	8D30: 20 60 CA A8 01 B1 BC 29(B2)	1008: D6 20 85 D7 A5 D6 79 CE<5C>
0A60:85 08 A5 D6 85 D7 A5 05<4E>	WD38:97 AA C8 B1 8C A6 20 80<30>	1010:20 85 D6 A5 D5 79 C6 20 <fe></fe>
@A68:85 D6 86 D5 A5 D5 C9 10(54)	0040:CA 20 80 CA 86 ED 84 E3<1F>	1018:85 D5 88 10 CE 32 05 F0<49>
0A70:90 31 E6 D4 A5 D5 C9 10431>	0048:38 A5 8C E9 03 85 8C 86(89)	1020:03 88 18 C7 4C E8 C1 18(EF)
0A78:90 27 4A 66 D6 66 D7 66<8A>	0050:02 C6 80 C5 0E D0 86 A5(1B)	1028:60 38 60 A5 E0 F0 FA A5<04>
0A80:D8 66 D9 4A 66 D6 66 D7<08>	8058:80 C5 OF F0 15 A0 00 B1(A2)	1030:04 F0 F4 45 E0 29 88 85<32>
0A88:66 08 66 09 4A 66 06 66(18)	@D&@:BC 10 B& 30 GA E& BA 1B<2F>	1838: EE A5 E8 29 7F 85 E8 A5 <b2></b2>
0A90: D7 66 D8 66 D9 4A 66 D6(9D)	0D68:A5 DE 65 E1 85 DE 90 02(44)	1040: D4 29 7F 38 E5 E0 18 69(27)
0A98:66 D7 66 D8 66 D9 85 D5(3E)	0D70:E6 DF 60 C6 BA 38 A5 DE<62>	1048:40 30 DE 05 EE A6 20 08<71>
0AA0:60 C6 D4 A9 00 86 D9 26(58)	0070:E5 E1 85 DE 80 02 C6 DF <db></db>	1950: E8 65 E6 85 E7 85 E8 85<3C>
ØAA8: 08 26 07 26 06 26 05 2A(8A)	0D80:60 A0 00 B1 BC 29 7F B5<91>	1050:E9 85 EA 65 E0 85 DA 84 <dc></dc>
DARD: 06 D9 26 D8 26 D7 26 D6(0F)	6088:E3 C8 B1 BC 4A 4A 4A 85 <b6></b6>	1060: D4 A8 88 A5 E8 D9 C6 28(3F)
ØAB8:26 DS 2A 06 D9 26 D8 26(F1)	8098:ED 68 E4 E4 E8 98 83 A2<19>	1068: D0 28 A5 E1 D7 CE 20 D0 <af></af>
BAC0: 07 26 D6 26 D5 2A 86 D9<18>	0D98:90 C8 60 CA 10 03 A6 E4(19)	1070:21 A5 EZ D9 D6 28 D0 1A <fc></fc>
ØAC8:26 D8 26 D7 26 D6 26 D5 <c1></c1>	@DA@:88 60 01 DE 3D C0 05 1D <de></de>	1078:A5 E3 D9 DE 20 D0 13 A5<48>
8AD9: 2A 68 68 AS 8B D9 FB A5(62)	@DA8:C8 @5 91 DE 60 B1 DE 10 <be></be>	1080:E4 D9 E6 20 D0 0C A5 E5<66>
9ADB: 06 85 D7 F8 F5 A0 80 84<90>	8098:C8 85 50 C8 85 F8 84 AD<38>	1088: D9 EE 20 D0 05 A2 00 4C(BC)
	9088:C8 95 68 AD C8 85 F6 83 <f3></f3>	
0AE0:08 84 DC 81 D4 10 92 C6<92>	CONTRACTOR OF AN ON TO USE EM ON FET	1090:02 E8 98 24 A5 E5 F9 EE <f1></f1>
ØAE8:08 20 54 C8 0A 0A 85 9C<86>	8DC0:A9 80 60 A9 01 60 20 FB<77>	1898:20 85 E5 A5 E4 F9 E6 20<01>
	8DC0:A9 80 60 A9 01 60 20 FB<77>	
0AF0:A9 00 2A 06 BC 2A 6D F4<50>	8DC0:A9 90 60 A9 01 60 20 FB<77> 8DC8:F7 AD FE 02 48 8E FE 02 <df></df>	10A0:85 E4 A5 E3 F7 DE 20 65<13>
0AF0:A7 00 2A 06 BC 2A 6D F4<50> 0AF8:02 85 BD 20 9E C8 80 D2<47>	8DC0:A9 90 60 A9 01 60 20 F8<77> 0DC8:F7 AD FE 02 48 8E FE 02 <af> 0DD0:A9 7D 20 F4 ED 68 8D FE<a3></a3></af>	10A0:85 E4 A5 E3 F7 DE 20 05<13> 10A0:E3 A5 E2 F7 D6 20 05 EZ<60>
0AF0:A7 00 2A 06 8C 2A 6D F4<50> 0AF8:02 85 8D 20 9E C8 80 D2<49> 0E00:A0 08 84 DD A4 DC 81 8C<40>	8DC0:A9 90 60 A9 01 60 20 F8<77> MDC8:F7 AD FE 02 48 8E FE 02 <af> MDD0:A9 7D 20 F4 ED 68 8D FE<a3> MDD0:B2 60 C9 FF 98 02 C0 58&lt;28&gt;</a3></af>	10A0:85 E4 A5 E3 F9 DE 20 05<13> 10A0:E3 A5 E2 F9 D6 20 05 E2<68> 10B0:A5 E1 F9 CE 20 05 E1 A5 <cd></cd>
0AF0:A7 00 2A 06 BC 2A 6D F4<50> 0AF8:02 85 BD 20 9E C8 80 D2<47>	8DC0:A9 90 60 A9 01 60 20 F8<77> 0DC8:F7 AD FE 02 48 8E FE 02 <af> 0DD0:A9 7D 20 F4 ED 68 8D FE<a3></a3></af>	10A0:85 E4 A5 E3 F7 DE 20 05<13> 10A0:E3 A5 E2 F7 D6 20 05 EZ<60>
0AF0:A7 00 2A 06 BC 2A 6D F4<50> 0AF8:02 85 BD 20 9E C8 80 D2<49> 0E00:A0 08 84 DD A4 DC 81 8C<40> 0B08:45 DB B5 DA A6 ED A4 E3 <f1></f1>	0DC0:A9 00 60 A9 01 60 20 F8<77> 0DC0:F7 AD FE 02 48 BE FE 02 <af> 0DD0:A9 7D 20 F4 ED 68 BD FE<a3> 0DD0:A9 AD C9 FF 90 02 CB 58&lt;28&gt; 0DE0:BA 69 00 85 D4 60 98 29&lt;9C&gt;</a3></af>	10A0:85 E4 A5 E3 F7 DE 20 05<13> 10A0:85 E4 A5 E2 F7 D6 20 05 E2<60> 10B0:A5 E1 F7 CE 20 05 E1 A5 <cd> 10B0:E0 F7 C6 20 05 E0 26 D5&lt;9E&gt;</cd>
0AF0:A9 00 2A 06 BC 2A 6D F4<50> 0AF8:02 05 BD 20 9E C8 80 D2<49> 0E00:A0 08 84 DD A4 DC 81 BC<40> 0E00:45 DB B5 DA A6 ED A4 E3 <f1> 0E10:06 DA B1 DE 3D C8 05 90&lt;16&gt;</f1>	8DC0:A9 00 60 A9 01 60 20 F8<77> MDC0:F7 AD FE 02 48 BE FE 02 <af> MDD0:A9 7D 20 F4 ED 68 BD FE<a3> MDD0:B2 60 C9 FF 90 02 CB 50&lt;28&gt; MD6:B2 60 C9 FF 90 02 CB 50&lt;28&gt; MD6:B2 64 A9 85 D4 60 98 29&lt;9C&gt; MDE8:F6 4A 85 D4 4A 4A 65 D4<af></af></a3></af>	10A0:85 E4 A5 E3 F7 DE 20 05<13> 10A0:E3 A5 E2 F7 D6 20 05 E2<60> 10B0:A5 E1 F7 CE 20 05 E3 E3 A5 <cd> 10B0:E0 F7 C6 20 05 E0 26 D5&lt;90&gt; 10C0:C3 C0 08 D0 9E A5 00 A5&lt;00&gt;</cd>
0AF0:A9 00 2A 06 BC 2A 6D F4<50> 0AF8:02 85 BD 20 9E C8 80 D2<49> 0F00:A0 08 84 DD A4 DC 81 8C<40> 0B08:45 DB B5 DA A6 ED A4 E3 <f1> 0B10:06 DA B1 DE 3D D8 05 90&lt;16&gt; 0B18:03 1D C8 05 91 DE C6 DD&lt;9E&gt;</f1>	8DC0:A9 00 60 A9 01 60 20 F8<77> MDC8:F7 AD FE 02 48 BE FE 02 <af> MDD0:A9 7D 20 F4 ED 68 BD FE<a3> BDD8:02 60 C9 F5 90 02 CB 50&lt;28&gt; MDE0:BA 69 00 85 D4 60 98 29&lt;9C&gt; MDE0:BA 69 00 85 D4 60 76 29&lt; MDE0:BS D4 98 29 0F A6 D6 E8<m2></m2></a3></af>	10A0:85 E4 A5 E3 F7 DE 20 85<13> 10A0:85 E4 A5 E2 F7 D6 20 85 E2<60> 10B0:A5 E1 F7 CE 20 85 E1 A5 <cd> 10B0:B5 F7 C6 20 85 E0 26 D5 10C0:C8 C8 08 D0 7E A6 00 A5 10C0:E1 D7 C6 20 D0 28 A5 E2&lt;94&gt;</cd>
0AF0:A7 00 2A 06 BC 2A 6D F4<50> 0AF8:02 95 BD 20 9E C8 80 D2<47> 0R00:A0 08 84 DD A4 DC 81 8C<40> 0808:45 DB B5 DA A6 ED A4 E3 <f1> 0810:06 DA B1 DE 3D C8 05 90&lt;16&gt; 0818:03 1D C8 05 91 DE C6 DD&lt;96&gt; 0820:F0 07 28 BD CA C4 E1 98&lt;7D&gt;</f1>	8DC0:A9 80 60 A9 81 60 28 F8<77> 8DC8:F7 AD FE 82 48 8E FE 82 <df> 8DD0:A9 7D 20 F4 ED 68 8D FE<a3> 8DD8:82 60 C9 FF 90 82 CB 58&lt;28&gt; 8DE0:8A 69 88 D 4 60 98 29&lt;9C&gt; 8DE0:BA 69 80 85 D4 60 98 29&lt;9C&gt; 8DE0:B D4 98 29 8F A6 D6 EB&lt;82&gt; 8DF8:50 45 D4 85 D4 60 A2 88&lt;7B&gt;</a3></df>	10A0:85 E4 A5 E3 F7 DE 20 85<13> 10A0:85 E4 A5 E2 F7 D6 20 85 E2<60> 10B0:A5 E1 F7 CE 20 85 E1 A5 <cd> 10B0:A5 E1 F7 CE 20 85 E1 A5<cd> 10B0:B0:B0 F7 C6 20 85 E0 26 D5&lt;9E&gt; 10C0:C8 C0 80 D0 9E A6 00 A5&lt;0B&gt; 10C0:E1 D7 C6 20 D0 28 A5 E2&lt;94&gt; 10D0:D7 CE 20 D0 21 A5 E3 D7&lt;0C&gt;</cd></cd>
0AF0:A9 00 2A 06 BC 2A 6D F4<50> 0AF8:02 85 BD 20 9E C8 80 D2<49> 0F00:A0 08 84 DD A4 DC 81 8C<40> 0B08:45 DB B5 DA A6 ED A4 E3 <f1> 0B10:06 DA B1 DE 3D D8 05 90&lt;16&gt; 0B18:03 1D C8 05 91 DE C6 DD&lt;9E&gt;</f1>	8DC0:A9 00 60 A9 01 60 20 F8<77> MDC8:F7 AD FE 02 48 BE FE 02 <af> MDD0:A9 7D 20 F4 ED 68 BD FE<a3> BDD8:02 60 C9 F5 90 02 CB 50&lt;28&gt; MDE0:BA 69 00 85 D4 60 98 29&lt;9C&gt; MDE0:BA 69 00 85 D4 60 76 29&lt; MDE0:BS D4 98 29 0F A6 D6 E8<m2></m2></a3></af>	10A0:85 E4 A5 E3 F7 DE 20 85<13> 10A0:85 E4 A5 E2 F7 D6 20 85 E2<60> 10B0:A5 E1 F7 CE 20 85 E1 A5 <cd> 10B0:B5 F7 C6 20 85 E0 26 D5 10C0:C8 C8 08 D0 7E A6 00 A5 10C0:E1 D7 C6 20 D0 28 A5 E2&lt;94&gt;</cd>

Listing 2. Turbo-Basic-Compiler-Runtime-Paket (Fortsetzung)



10E0:20 D0 13 AS E5 D9 E6 20<2F>	1388: D6 85 E2 A5 D7 85 E3 A5(E9)	1690:D0 CB 81 FC 05 D9 60 85<45>
18E8:08 8C A5 E6 D9 EE 28 D04F5>	1300:00 85 E4 A5 D9 B5 E5 A9(5A)	1698:FC 84 FD A0 00 81 FC 9D(E5)
10FB: 05 A2 01 4C 02 E8 90 2A<0C>	13C8:00 85 D4 85 D5 85 D6 85(C0)	16A0:00 D8 C8 B1 FC 9D 00 D9<12>
18F8:A5 E6 F9 EE 28 85 E6 A5(E6)	13D0: D7 85 D8 85 D9 60 68 85<84>	16A8: C8 81 FC 9D 88 DA C8 B1 <e6></e6>
		1680:FC 9D 00 DR CB 91 FC 9D<55>
1100:E5 F9 E6 20 85 E5 A5 E4<6E>	1308:FC 68 85 FD 66 FC 00 02(F9)	
1108:F7 DE 28 85 E4 A5 E3 F9 <cc></cc>	13E0:E6 FD AZ 80 A1 FC F0 07 <c8></c8>	1688:00 DC C8 B1 FC 9D 00 DD<1F>
1110:D6 20 05 E3 A5 E2 F9 CE<70>	13E8: A8 28 81 21 4C 1E E9 A5 <e4></e4>	16C0:60 9D 00 D9 A9 40 9D 00kD8>
		16C8: D8 A9 00 9D 00 DA 9D 00<77>
1118:20 85 E2 A5 E1 F9 C6 20 <c8></c8>	13F0:FD 48 A5 FC 48 60 C0 00<5E>	
1129:85 E1 26 D6 C8 C8 88 D8<2C>	13F8:10 FB 98 85 C3 D0 A8 00 <fc></fc>	1600:08 90 00 DC 90 00 DD 60<5F>
1128:9E A0 80 A5 E2 D9 C6 20(79)	1400:84 93 BC FE 02 C9 80 D0<66>	1608:85 D5 A9 40 85 D4 A9 00<1A>
1130:00 28 A5 E3 D9 CE 20 D0 <f5></f5>	1408:02 84 11 A2 FD 9A 68 85<02>	16E0:85 D6 85 D7 85 D8 85 D9(CF)
1138:21 A5 E4 D9 D6 20 D0 1A(16)	1418:D2 68 85 D3 48 A5 DZ 48(89)	16E8:60 20 1A EC 86 D3 20 96<9A>
1140:A5 E5 D9 DE 20 D0 13 A5 <f2></f2>	1418:A5 82 85 D6 A5 83 85 D7(3A)	16F0:E8 20 FC F1 A6 D3 4C 51(92)
1148:E6 D9 E6 20 D0 0C A5 E7<31>	1420: A5 98 D0 3F A0 00 C8 C8(A5)	16F8: EB 85 FC 84 FD AM BM B1 <ca></ca>
		1700:FC 65 E0 C8 B1 FC 85 E1<25>
1150: D9 EE 20 D0 05 A2 02 4C(SA)	1428:B1 D6 C5 D2 C8 B1 D6 E5<65>	
1158:02 E8 90 2A A5 E7 F9 EE(B2)	1430:D3 B8 87 C8 D9 F8 E6 D7<2C>	1708: C8 81 FC 85 E2 CB B1 FC(AB)
1160:20 BS E7 A5 E6 F9 E6 20<2A>	1438: DØ EC 38 AS D6 E9 06 85<77>	1710:85 E3 C8 81 FC 85 E4 C8<13>
1168:85 E6 A5 E5 F9 DE 20 85 <ce></ce>	1440: D6 80 02 C6 D7 B1 D6 B5 <c6></c6>	1718:81 FC 85 E5 40 85 FC 84<0D>
1170:E5 A5 E4 F9 D6 20 85 E4<96>	1448:88 88 81 D6 85 BA A5 BC<70>	1720:FD A0 00 A5 E0 91 FC CB(50)
1178:A5 E3 F9 CE 20 85 E3 A5(9F)	1450:A4 80 30 OF A2 80 86 BD<7F>	1728:AS E1 91 FC CB A5 E2 91(FC)
1180:E2 F9 C6 20 85 E2 26 D7<5B>	1458:E4 C3 D0 04 A6 95 F0 03<60>	1730:FC C8 A5 E3 91 FC C8 A5 <c2></c2>
1188:C8 C0 06 D0 9E A0 00 A5 <cd></cd>	1460:4C 42 F8 20 18 E9 FD 98(38)	1738:E4 91 FC C8 A5 E5 91 FC(96)
1190:E3 D9 C6 20 D0 28 AS E4<5F>	1468:46 65 68 6C 65 7Z 28 ZDKEE>	1740:60 20 77 EB 4C CE ED B6<80>
1198:D9 CE 20 00 21 AS E5 D9 <de></de>	1470:20 00 A5 C3 A0 00 20 B1<7F>	1748:D3 48 20 77 EB 68 20 F9<9F>
11A0: D6 20 D8 1A A5 E6 D9 DE<97>	1478:EC 20 18 E9 20 69 6E 20 <b3></b3>	1750:F1 A6 D3 4C 51 EB 85 D4 <aa></aa>
11A8:20 D0 13 A5 E7 D9 E6 20<13>	1480:5A 65 69 6C 65 20 90 A5<63>	1758:84 D5 A0 00 81 D4 85 D6 <ee></ee>
11E0:00 0C A5 E8 09 EE 28 00<67>	1488:BA A4 BB 20 81 EC 20 18<95>	1760:84 D7 E6 D4 D0 02 E6 D5 <af></af>
1188:05 A2 03 4C 02 E8 90 2A(E5)	1490:E9 20 28 24 00 38 A5 D2<78>	1768:60 20 90 EC 4C 87 EC 20(B5)
11C0:A5 E8 F9 EE 20 85 E8 A5(R5)	1498:E9 02 48 A5 D3 E9 00 20:68>	1770:13 CØ 20 21 F4 A6 93 A5<30>
11CB;E7 F9 E6 20 85 E7 A5 E6(52)	14A0:94 EA 68 28 94 EA A9 29 <dc></dc>	1778: D4 9D 44 03 A5 D5 9D 45(8F)
1100:F9 DE 20 85 E6 A5 E5 F9<97>	14A8: 20 F4 ED A9 88 B5 93 28<5F>	1780:03 A5 D6 9D 48 63 18 65(C9)
11D8:D6 20 85 E5 A5 E4 F9 CE<61>	1488:18 E9 98 58 72 AF A7 72 <fd></fd>	1788:94 85 94 A5 D7 9D 49 B3(8F)
11E8:20 85 E4 A5 E3 F9 C6 20(E0)	14B8:61 6D 6D 65 6E 64 65 28(4A)	1790:05 D6 F0 05 A9 8B 4C C6<15>
11E8:85 E3 26 DB C8 C0 98 D0(E5)	1400:3A 20 C4 6F 73 2C 20 D2<53>	1798: EA 60 4C 61 FA 85 99 B4 <ba></ba>
11F8:9E A8 88 A5 E4 D9 C6 20(C3)	1408: 75 6E 20 6F 64 65 72 20(2F)	17A0:9A 18 45 90 A8 A5 91 45(A1)
11F8: DØ 28 A5 E5 D9 CE 20 D0<00>	14D0:CC 6F 61 64 29 3F 80 A9<84>	17AB: 9A CD C6 02 90 09 D0 EACDF>
1200:21 A5 E6 D9 D6 20 D0 1A<18>	14D8:40 8D BE 07 A7 00 8D 86<51>	1780:CC 65 02 90 02 D0 E3 85<11>
1208:A5 E7 D9 DE 20 D0 13 A5<90>	14E0:02 20 D1 21 C9 44 F8 08<78>	1788: 0F 84 0E 38 A5 90 E5 8E<53>
1210:E8 D9 E6 28 D0 0C A5 E9 <fe></fe>	14E8: C9 4C F0 8A C9 52 D0 E747C>	1700:85 BC AS 91 ES BF 85 BD <ad></ad>
1218:09 EE 28 00 05 A2 84 4C<2B>	14F0:4C 80 20 4C 84 28 20 18(3D)	1708:18 A5 BE 85 97 85 88 65<42>
1220102 E8 90 2A A5 E9 F9 EE(8C)	14F8:E9 98 46 69 60 65 6E 61(EF)	1700:99 85 8A 85 8E A5 BF 85KE7>
1228:20 85 E9 AS EB F9 E6 20<34>	1500:60 65 20 44 3A 00 20 4A(CE)	1708:98 85 89 65 9A 85 8B 85(9E)
1238185 E8 A5 E7 F9 DE 28 85<98>	1508:F1 A0 FF C8 B1 F3 C9 98(7E)	1760:8F A5 0E 85 90 A5 0F 05(F3)
1238:E7 A5 E6 F9 D6 20 85 E6(A5)		17E8:91 4C E1 21 29 0F 06 BA(2B)
	1518:F0 29 C9 3A D0 F5 C8 B1(D4)	
1240: A5 E5 F9 CE 20 B5 E5 A5<71>	1518:F3 C9 98 D0 F9 A9 2E 91 <a6></a6>	17F0:06 8A 06 8A 06 8A 05 BA(DB)
1248:E4 F9 C6 28 85 E4 26 09<30>	1528:F3 C8 A9 43 91 F3 C8 A9(C8)	17F8: A6 88 E0 05 90 03 4C 29(F1)
1250:CB C0 08 D0 9E A5 D5 D8<3F>	1528:54 91 F3 C8 A9 42 91 F3(FE)	1800:F5 9D C4 02 60 A0 78 06(E4)
1258:64 AG GO A5 E5 D9 C6 20<9E>	1530:C8 A5 F3 A6 F4 28 53 F4 <de></de>	1808:80 20 A0 00 80 08 D2 A2<1F>
1260: D0 28 45 E6 D9 CE 28 D0< FA>	1538:4C 8B F9 CB 89 7F 85 99(ED)	1810:03 8E 0F 02 06 9C A6 8C(9E)
1268:21 A5 E7 D9 D6 28 D0 1A(A1)	1540:81 85 88 00 F7 A9 44 8D<69>	1818:00 08 00 E2 05 BC A5 BB(83)
1270:A5 E8 D9 DE 28 D8 13 A5<85>	1548:88 85 A7 3A 8D 81 85 4CKEC>	1820:90 00 02 98 F0 07 EB EB(A5)
1278:E9 09 E6 28 00 0C A5 EAKE9>	1550: 49 EA 48 4A 4A 4A 4A 20 <cf></cf>	1828: A5 89 9D 00 D2 A5 8A 0A <a6></a6>
1288:09 EE 28 08 85 A2 85 4C(A2)	1558: 9F EA 68 29 0F 09 38 C9<32>	1838:0A 0A 0A 05 8C 9D 01 D2<24>
1288:02 E8 98 2A A5 EA F9 EE<09>	1568:3A 98 82 69 86 4C F4 ED(88)	1838:60 38 24 18 20 BF ED 4C<4E>
1298:20 85 EA A5 E9 F9 E6 20(D4)	1568:A2 78 20 C4 EA BA 38 E9(F5)	1840:8A ED 38 24 18 20 C6 ED(73)
1298:85 E9 A5 E8 F9 DE 28 85(01)	1570: 18 AA DO F6 AZ OB AF 00(E4)	1848: A5 C8 80 FB 02 A2 A0 A9<3E>
12A0:E8 A5 E7 F9 D6 20 B5 E7 <bd></bd>	1578:90 00 D2 CA 10 FA 68 20<91>	
	1378: YD 98 DZ CR 10 FR 56 26 T17	1858:0C 9D 4A 03 A9 00 9D 4B<32>
12A8:A5 E6 F9 CE 20 B5 E6 A5<1C>	1580:EA EE A9 0C 9D 42 83 20(0C)	1858:03 A9 11 69 00 4C C6 EA(D5)
12B0; E5 F9 C6 20 85 E5 26 DA(36)	1588:A8 20 30 01 60 4C 3C E9(83)	1860:48 A9 86 20 C1 EA 68 A8(BE)
1288: C8 C0 08 00 9E 4C E8 C1 <a9></a9>	1598:A6 92 E6 92 A5 D4 9D 80KBF>	1868: 29 FØ 49 1C AA A9 ØD 86 <a4></a4>
12C0:36 D5 C8 C0 08 D0 F9 F0<98>	1598: DE A5 D5 9D 2A DE A5 D6<31>	1970:98 84 8A A2 21 A0 82 28 <dc></dc>
12C81F4 F8 18 A5 ES 80 F5 20<99>	15A8:90 54 DE AS D7 9D 7E DE<36>	1878:53 F4 4C 4A EE 85 55 84<31>
12D0:65 E5 8D F4 20 A5 E4 8D<39>	15A8: A5 D8 90 A8 DE A5 D9 90 <fd></fd>	1880:56 86 54 60 85 54 98 FØ(B2)
12DB:ED 20 65 E4 8D EC 20 A5<3F>	1580:02 DE 60 20 80 C2 C6 92<13>	1888:0C 4C 29 F5 20 F9 CA A5<23>
12E0:E3 80 E5 20 65 E3 80 E4<65>	[588:A6 92 BD 00 DE 85 D4 BD<18>	1890: D4 A4 05 80 F4 60 20 BF<49>
12E8: 20 A5 E2 80 DD 20 65 E2<6C>	15CB: 2A DE 85 D5 BD 54 DE 85 <c9></c9>	1898:ED AC E1 ED 28 C6 ED AZ<98>
12F@:BD DC 20 A5 E1 8D 05 20<28>	1508: D6 BD 7E DE 85 D7 BD A8<65>	18A0:60 A4 C8 20 B1 21 4C 38<7F>
12F8:65 E1 8D D4 20 A9 00 8D<26>	1500:DE 85 DB 80 D2 DE 85 D9<80>	18A8: E9 A9 00 85 93 A9 98 2C(A4)
1300:CD 20 69 00 BD CC 20 A2<1F>	15DB:68 C6 92 A6 92 BD 88 DE(B9)	1880: A9 20 A6 93 A8 E6 94 20<49>
1308:02 BD F2 20 7D F2 20 9D<27>	1588:85 E0 80 2A DE 85 E1 80(A0)	1989: 81 21 4C 38 E9 20 C6 ED <d8></d8>
1310:F1 20 BD EA 20 7D EA 20<07>	1569:54 DE 85 E2 BD 7E DE 85(84)	1800: A9 60 85 93 86 88 20 69(0A)
1318:90 E9 28 80 E2 28 70 E2<83>	15F8:E3 BD A8 DE 85 E4 80 D2 <cc></cc>	1909:EF A0 00 A6 88 40 4A EB(A3)
1320:20 9D E1 20 8D DA 20 7D <e6></e6>	15F8: DE 85 E5 60 A6 92 E6 92 (D6)	1800:86 88 20 81 EF A0 00 A6<11>
1328:0A 20 9D 09 20 8D D2 20(FD)	1600:9D 00 DE 98 9D 2A DE 60(E1)	1808:88 4C 4A EB 86 88 A0 07(C7)
1330:70 D2 20 9D D1 20 8D CA<21>	1608:86 D3 20 13 CD A6 D3 A5<48>	18E0: 20 2C EE A6 88 4C 51 EB<42>
1338:20 7D CA 20 9D C9 20 CA <e3></e3>	1610:D4 9D 80 D8 A5 D5 9D 80<94>	18E8: A0 08 A9 06 A6 93 9D 48(A3)
1348:D0 C7 AD F4 20 6D F2 20(AF)	1618:D9 AS D6 9D 00 DA A5 D7(F6)	18F0:03 A9 D4 9D 44 03 A9 00<91>
1348:80 F1 20 AD EC 20 60 EA(72)	1620:7D 00 DB A5 DB 7D 00 DC(55)	18F8:9D 45 03 9D 49 03 98 4C<05>
1350:20 BD E9 20 AD E4 20 60<32>	1628: A5 D9 9D 00 DD A5 11 F0<02>	1900:C6 EA 86 93 85 88 84 8A<58>
1358: E2 28 80 E1 20 AD DC 20<9E>	1630:01 60 4C 54 FE BD 00 DB(6E)	1988: 49 83 85 BC A6 93 A5 BC(OF)
1360:60 DA 20 80 D9 20 AD D4 <fd></fd>	1638:85 D4 BD 00 D9 85 D5 BD<74>	1910:9D 42 83 A5 88 9D 4A 83<26>
1368:20 60 D2 20 6D D1 20 AD <c9></c9>	1648:00 DA B5 D6 BD 00 D8 B5<19>	1918: A5 8A 9D 48 03 A5 D4 9D<78>
1370:CC 20 60 CA 28 80 C9 20<01>	1648:D7 8D 00 DC 85 D8 8D 00(CB)	1920:44 03 A5 D5 9D 45 03 20<33>
1378:A2 02 80 EF 20 70 EF 20(E7)	1650:00 85 D9 60 BD 00 D8 85<47>	1929: A1 EF 20 AB 20 20 BA EF <f9></f9>
1380:90 EE 20 BD E7 20 7D E7<5E>	1658:E0 8D 00 D7 85 E1 8D 00 <da></da>	1930:8C 43 83 4C 38 E9 86 88<2E>
1388: 20 90 E6 20 90 DF 20 70 <c3></c3>	1660:DA B5 E2 BD 00 D9 B5 E3(7C)	1938: A9 00 A6 93 9D 42 03 20 <e3></e3>
1390: DF 20 9D DE 20 0D D7 20<60>	1668:8D 00 DC 85 E4 BD 00 DD<26>	1940:AB 20 98 A0 00 A6 B8 4C <c2></c2>
1398:70 07 28 90 06 28 BD CF<7E>	1670:85 E5 60 85 FC 84 FD A0<68>	1948:4A EB 86 88 84 8A A9 26(C4)
13A8:28 7D CF 28 9D CE 28 8D<33>	1678:00 B1 FC 85 D4 C8 B1 FC <cd></cd>	1950: A6 93 20 C6 EA BD 4C 03 <f8></f8>
13A8:C7 20 70 C7 20 90 C6 20<55>	1689:85 05 C8 B1 FC 85 D6 C8(32)	1958: BC 4D 03 A6 88 28 4A EB(31>
1380:CA 10 C7 A5 D5 B5 E1 A5<06>	1688:B1 FC 85 D7 C8 B1 FC 85(AC)	1968:A6 93 BD 4E 03 A0 00 A6<43>
1298:FM 16 FL M3 N2 R3 F1 M3/69)	1990:01 LC 93 A\ C9 BT LC 93(HC)	1700; MO 70 BU 4E 60 MU 6M A6(45)

Listing 2. Turbo-Basic-Compiler-Runtime-Paket (Fortsetzung)

1968:8A 4C 4A EB 4C 29 F5 A6<81>	1C40: AC E6 F2 A6 88 4C 51 EB(61> 1	1F18:00 85 D7 60 A9 3F 85 D4 <c1></c1>
1970:93 9D 4E 03 98 D0 F5 A5(18)	1C48:A9 80 85 F2 F8 F5 20 44 <dc></dc>	1F20: A2 Ø5 AD ØA D2 29 FØ C9(82)
1978:88 9D 4C 83 A5 89 9D 4D(A1)	1C50:F1 4C 99 F1 20 34 F1 18 <ff></ff>	1F29: A0 90 F7 85 DA AD 8A D2(C4)
1980:03 A9 25 4C C6 EA A9 07<05>	1C58: A5 F3 65 F2 85 F3 98 82<88>	1F30:29 0F C9 0A 80 F7 05 DA<55>
1988:20 C1 EA A2 88 A0 00 A9(FA)		
1990:10 4C B1 ED A9 80 20 C1(09)	1C60:E6 F4 A0 00 B1 F3 C9 98 <ia></ia>	1F38:95 D4 EA DØ E5 4C EC C1<19>
	1C68:F0 03 C8 D0 F7 A5 F3 A6<35>	1F40: 20 80 C2 20 SE F4 20 0E<85>
1998:EA A9 C0 B5 6A A2 0C A0 <cc></cc>	1070:F4 20 53 F4 A6 88 4C 74 <ae></ae>	1F48:F2 4C C1 F4 AS D4 29 7F<77>
19A0:00 A9 13 4C B1 ED A9 00 <de></de>	1078:FE A5 D4 F0 04 49 80 85(IB)	1F50:38 E9 40 90 11 AA A9 00<66>
19A8:C9 08 B8 14 0A 0A 0A 0A 0A<	1CB0:04 28 18 EB 4C FC F1 29<10>	1F58:E0 05 90 02 A2 84 95 D5<1D>
1988:85 93 AA A9 88 85 94 85(68)	1C88:18 EB 4C 0E F2 20 F5 EAK57>	1F40:CA 10 FB 4C EC C1 40 20 <ee></ee>
1988:F2 A4 C9 D0 02 E6 C9 60<7C>	1090:40 17 F2 20 80 C2 40 FC<00>	1F48:17 F2 A5 D4 29 7F 38 E9(1F)
1908: A9 14 40 30 E9 A9 87 A2<47>	1C98:F1 20 B8 C2 4C 0E F2 A0<81>	1F70:40 90 44 AA A9 00 E0 04<1F>
1908:21 A0 06 20 53 F4 A2 70(0D)	1CA0: CO DO 84 A9 61 A8 40 85(43)	1F78:80 3A 95 06 ER 90 F7 A5<21>
19D8: A9 86 A0 00 20 44 EE 20<5A>	1CAB: E1 84 EØ A7 80 85 E2 85<57>	1F80: D4 29 7F 38 E9 40 90 2D <dc></dc>
1908: 6F EF CO 88 FO OB 20 38(60)	1C80:E3 85 E4 85 E5 F0 03 20<95>	1F88: AA EØ Ø4 BØ D9 A9 ØØ AB<30>
19E8:E9 A2 80 20 F6 ED 4C 19<36>	1CB8:3B EC 20 D1 C0 B0 1C 60 3	1F98:15 D6 94 D6 E8 E8 64 D8<45>
19E8:EF A2 70 4C C4 EA A9 20<24>	1CD0:28 38 EC 29 CB CB 80 13 <a0></a0>	
	1CCB:60 20 3B EC 20 0B E4 80(03)	1F98:F7 24 D4 10 17 A8 F0 14<22>
19F8: 2C A9 23 2C A9 24 2C A9<68>		1FA0: A7 C0 85 E0 A6 01 84 E1 <c4></c4>
19F8:21 A2 70 86 93 A0 00 84<38>	1CD0:0A 60 20 3B EC 20 6D E5<7C>	1FA8:88 84 E2 84 E3 84 E4 84<78>
1A00:88 84 8A 4C 4C EE A2 0B(AF)	1CDB: BØ Ø1 60 A9 ØB 4C 3D E9<38>	1F98:E5 4C D1 CØ 60 Ø6 D4 20<6D>
1A08:2C A2 07 48 8A A6 93 90<01>	1CE0:85 E1 A9 40 85 E8 A9 00<25>	1FBB:09 E9 90 FB B0 E2 20 5E <ca></ca>
1A10:42 03 A5 88 9D 44 03 A5 <f4></f4>	1CE8:85 E2 85 E3 85 E4 85 E5 <c3></c3>	1FC0:C5 BØ 24 60 20 68 C5 BØ <ad></ad>
1A18:89 9D 45 03 69 9D 48 03(F9)	1CF0:F0 E3 85 E1 A9 40 85 E0 <a6></a6>	1FC8: 1E 60 20 26 CA 80 18 60(10)
1A20:98 90 49 03 4C C9 EA 20(1B)	1CF8: A7 00 85 E2 85 E3 85 E4 <f8></f8>	1FD8: 20 52 C4 B8 12 68 28 56(CA)
1A28:6F EF 4C 38 E9 A6 93 A9<0A>	1000:85 E5 F0 C8 85 E1 A9 40(F3)	1FDB: C4 BØ ØC 60 20 72 C3 BØ <ec></ec>
1A30:00 9D 4B 03 9D 49 83 A9(7D)	1008:85 ER A9 00 85 E2 85 E3(71)	1FE0:06 60 20 91 C6 90 FA A9<53>
1A38:07 90 42 83 4C AB 20 20 <f4></f4>	1D10:85 E4 85 E5 60 A5 D4 F0(B3)	1FE8:03 4C 3D E9 A5 E0 30 F7 <d4></d4>
1A40:D1 21 4C 39 E9 A5 C9 B5(16)	1D18:04 49 80 85 D4 60 A5 D4<4C>	
		1FF8:4C 09 E9 4C DD C3 20 3B(ED)
1A48:88 A4 74 C8 C4 88 70 05(E2)	1020:29 7F B5 04 60 20 3B EC<60>	1FF8:EC A5 FØ FØ F6 A5 D4 FØ<2E>
1A50:18 65 C9 90 F2 20 F2 ED(EA)	1D28:20 03 F3 90 S5 F0 53 80(5B)	2000: EB 10 26 29 7F 85 D4 A5<5F>
1A58:A5 94 C5 88 DØ F7 AØ A4<32>	1D30:4E 20 3B EC 20 03 F3 F0<0A>	2008:E0 29 7F 3B E9 40 90 D7 <f4></f4>
1A60:D6 A5 D7 F0 02 A0 FF 84<13>	1D38:46 D8 47 20 3B EC 20 03<00>	2010:A2 04 C9 04 F0 0E B0 11(C3)
1A68:88 B1 D4 85 8A A5 D4 85(EF)	1D40: F3 98 3F B0 3A 20 3B EC(6F)	2018: AA AS B9 62 00 D0 CB CB(E7)
1A70:80 A5 D5 B5 BD A9 9B 2C<5F>	1D48:20 03 F3 90 32 F0 30 B0<1C>	2020: CO 04 DO F6 B5 E1 4A B0(2A)
1A78: A5 8A A4 88 91 8C 60 18(88)	1050:31 20 38 EC 20 03 F3 90<04>	2028:01 18 A9 00 6A 48 A5 E0<9F>
1AB0:24 38 6E 16 21 A9 3F A0<80>	1058:26 BØ 27 20 38 EC 20 03<45>	2030: C7 40 00 40 A5 E2 65 E3(5D)
1ABB: 21 BD E0 02 BC E1 02 A2<52>	1D60:F3 F0 1F D0 1A 20 3B EC <a4></a4>	
		2038:05 E4 05 E5 D0 43 A5 E1<9A>
1A90: 18 A9 84 A0 80 20 44 EE (AB)	1068: A5 E0 25 D4 F0 11 D0 12 <d4></d4>	2040:29 F0 4A 85 FC 4A 4A 65 <a3></a3>
1A98: 20 69 EF C9 FF D0 SC 20(B1)	1D70:20 3B EC AS E0 05 D4 F0 <bd></bd>	2048:FC 85 FC A5 E1 29 0F 65(E0)
1AA8:69 EF C9 FF D8 55 A7 3F(52)	1D78:06 D0 07 A5 D4 F0 03 4C<05>	2050:FC 85 FC 20 C9 C2 46 FC(3F)
1AA8: A0 21 80 E2 02 8C E3 02 (AE)	1080:09 E9 4C DD C3 A5 D4 F0 <f9></f9>	2058:80 03 20 00 C3 20 E8 C2 <ad></ad>
1AB0:20 69 EF 9D 44 03 20 69<86>	1088:F6 10 F7 20 C4 F2 A9 C0<5B>	2060: AS FC F0 50 20 26 C3 20(B4)
1AB8: EF 9D 45 03 3D 44 03 C9<2E>	1090:85 D4 60 20 3A F3 90 EA <a6></a6>	2068:08 E4 B0 OF 46 FC 70 F7(E6)
1AC0:FF F0 E3 20 69 EF 48 20<7C>	1098:F8 E8 B8 E3 20 3A F3 F8(C8)	2070:20 C9 C2 20 51 C3 20 0B(4E)
1AC8:69 EF A8 68 38 FD 44 Ø3(A9)	1DA0: DE D0 DF 20 3A F3 90 DA(72)	2078:E4 90 E2 4C 1D F2 4C 29 <f9></f9>
1AD0:90 48 03 99 FD 45 03 9D(6E)	1DAS: 80 05 20 3A F3 90 00 F0 <fd></fd>	2080:F5 A5 E5 48 A5 E4 48 A5<6F>
1AD8:49 03 FE 48 03 D0 03 FE<11>	1088: CE 88 CF 28 3A F3 98 C7<50>	2088:E3 48 A5 E2 48 A5 E1 48(A9)
1AE0:49 03 20 C9 EA 20 99 20<2B>	1DB8:80 C8 20 3A F3 F0 C3 D8<35>	
		2070:A5 E0 48 20 56 C4 B0 E6<07>
1AE8: A2 10 DE 43 03 F0 87 20 <b6></b6>	1DC0: BE AS D4 45 E0 D0 25 A5<0F>	2098:68 85 E0 68 85 E1 68 85<65>
1AF0:C4 EA 2C 16 21 10 03 4C <fa></fa>	1DC8:D5 C5 E1 D0 18 A5 D6 C5 <fb></fb>	20A0: E2 68 85 E3 68 85 E4 68(51)
1AFB: AZ 28 40 AF 01 4C 3D EF(84)	1000:E2 00 12 A5 07 C5 E3 00(6B)	20A8: B5 E5 20 00 E4 B0 CC 20(B2)
1800:20 4A F0 68 68 6C D8 00 <fb></fb>	1008:0C A5 DB C5 E4 D8 06 A5<00>	2080:81 C3 B0 C7 68 10 04 05 <f4></f4>
1808:85 D4 84 D5 A5 82 85 D6<83>	1DE0: D9 C5 E5 FB B6 6A 45 D4 <c7></c7>	2088:04 85 D4 60 85 E0 84 E1<22>
1B10:A5 B3 B5 D7 A0 00 B1 D6 <ad></ad>	1DE8:0A A9 01 60 30 05 A5 D4<22>	2000:20 FB EA 4C CE ED 20 FE<68>
1818: C5 D4 C8 81 D6 E5 D5 B8(67>	1DF0:C3 E0 60 06 E0 A9 01 60(FC)	2008:F5 25 E8 AA 98 25 E1 A8(25)
1829:09 C8 C8 C8 D0 F0 E6 D7<23>	1DF8: A0 00 A5 D6 D0 06 A5 D7(CC)	2000:8A 60 20 FE F5 05 E0 AAK76>
1929: DØ EC DØ 13 88 91 D6 C5<ØØ>	1E00:F0 05 C6 D7 C6 D6 AA 08<7C>	2008:98 05 C1 A8 8A 60 20 FE(58)
1830: D4 D0 0C C8 C8 B1 D6 85<77>	1E08: A5 E2 D0 06 A5 E3 F0 20(22)	20F0:F5 45 E0 AA 98 45 E1 A8(C2)
1838: 08 C8 81 D6 85 D9 68 A9(BF)	1E10:C6 E3 C6 E2 28 F0 16 B1(A8)	20E8:8A 60 9A 98 9D 9B 83 B5<83>
1840:09 4C 30 E9 A9 98 A8 B5<1D>	1518: D4 D1 E0 D0 05 E6 D4 D0 <eb></eb>	20F0: 80 82 A4 3C 7C 8C 27 67 <af></af>
1848:D4 84 D5 98 30 29 A5 89<5C>	1E20:02 E6 D5 E6 E0 D0 D3 E6 <ab></ab>	20F8:A7 85 04 A5 14 A6 11 F0 <fb></fb>
1850:A4 81 84 87 85 86 A8 00<95>	1E28:E1 DØ CF BØ Ø2 C8 18 6Ø(6D)	2100:10 C5 14 F0 F8 A5 D4 D0<08>
1858:81 86 C5 D4 C8 81 86 E5(A3)	1E30:28 39 60 38 AD E5 02 E5	2108:03 B8 30 05 C6 D4 4C 3D<95>
1860:D5 80 08 C8 81 86 65 86 <fb></fb>	1E38:0E AA AD E6 02 E5 0F A8(63)	2110:F6 60 20 C9 C2 20 07 C3<55>
1868:90 EA E6 87 80 E6 A5 86(8F)	1E48:8A 68 28 A1 EF A5 D4 A4<8D>	2:18:20 A9 F4 20 80 C2 20 2A<03>
1B70:D0 02 C6 B7 C6 B6 60 4C <b9></b9>	1E48:D5 B5 F3 84 F4 A0 00 84(C9)	2120:C3 20 0E F2 20 B0 C2 20K11>
1878:29 F5 A9 06 4C 3D E9 86(BD>	1E50:F2 60 20 84 F3 20 65 22(CC)	2128:26 C3 4C 05 F2 A2 5F 2C(10)
1880:88 A0 00 B1 86 C9 98 F0<72>	1E58:90 34 20 D0 F3 A7 12 4C<95>	2130:A2 FF 86 DF 85 DA 84 DB<37>
1888:08 E6 86 D0 15 E6 87 D0<49>	1E60:3D E9 20 84 F3 84 D4 84(87)	2138:20 18 EB 20 FB EA 18 A5<64>
	1E68:D5 B1 F3 38 E9 38 90 1E(A2)	2140: D4 65 DA B5 D4 A5 D5 65(DD)
1890:11 A0 03 B1 86 F0 E3 18(B4)		
1898:A5 86 69 84 85 86 98 82<51>	1E70:C9 17 80 1A C9 0A 90 06<03>	2148: D8 85 05 80 22 38 A5 06<48>
18A8:E6 87 A0 86 81 86 C9 2C(1C)	1E78: E9 07 C9 0A 90 10 A2 04(AB)	2150:E5 DA B5 D6 A5 D7 E5 DB <d8></d8>
18A8:F0 07 C9 98 F0 03 C8 D0<56>	1E80:06 D4 26 D5 CA D0 F9 05<48>	2158:85 D7 90 13 05 D6 F0 0F(A4)
1880:F3 A5 86 A6 87 20 53 F4<02>	1E88: D4 85 D4 C8 DØ DB 2Ø BA <aa></aa>	7160:38 A5 D6 E5 E2 85 DC A5<74>
1888:18 AS 86 65 D6 85 86 98 <a9></a9>	1890: EF 4C 00 CB A2 00 F0 0A(4A)	2168:D7 E5 E3 B5 DD B0 04 A9<4A>
18C0:02 E6 87 60 20 C1 F0 A6 <d2></d2>	1E98:A2 08 00 06 A2 0C DM 02(E8)	21/0:00 AS 60 E6 DA DØ 02 E6 <de></de>
1808:88 40 74 FE 20 C1 F0 A5(AD)	1EAB: A2 14 18 86 D4 65 D4 AAKA5>	2178: DB AØ ØØ B1 EØ 51 04 25<3C>
1800:04 A4 D5 A2 00 B5 F3 84<9A>	1EAB: BD 70 02 A0 00 60 4C 29 <dc></dc>	2180: DF DØ 30 A5 D4 85 D8 A5<7F>
1808:F4 86 F2 A5 D6 48 28 65(A7)	1EB8:F5 A8 88 A5 D5 F8 83 28(26)	2188: D5 85 D9 A5 E8 85 E4 A5(49)
18E0: 22 B0 0A 68 C5 F2 D0 05<60>	1EB8: FE F3 A5 D4 48 4A 4A 4A <ca></ca>	2196:E1 85 E5 A5 E3 B5 DE A6(26)
	1ECB: 4A 20 89 F4 68 29 8F 89<17>	
19E8: A6 88 4C 51 EB A9 08 4C<3B>		2198:E2 E9 CA D0 04 C6 DE 30(43)
1BF0:3D E9 86 88 A5 93 D0 0A<8D>	1EC8: 38 C9 3A 98 82 69 86 99 <f5></f5>	21A0:25 B1 D8 51 E4 25 DF D0(03)
18F8: A5 F2 D0 2A A9 3F 20 F4(54)	1ED0:80 05 C8 60 85 D4 84 D5 <de></de>	21A8: 8A CB D8 EE E6 D9 E6 E5(1E)
1000:ED 20 86 88 A5 F2 D0 1E<91>	1ED8: 28 F3 F3 A9 80 00 30 20 <ad></ad>	2120:4C DC F6 E6 D4 D8 82 E6(74)
1088:20 00 CB A6 93 A5 F3 90<54>	1EE0:58 22 A0 FF C8 B1 F3 10<31>	2188:05 A5 DC D0 04 C6 DD 30<29>
1C10:44 03 A5 F4 9D 45 03 A9 <c8></c8>	1EE8:FB 29 7F 91 F3 C8 A5 F3 <db></db>	21CB: AE C6 DC 4C 85 F6 A5 DA <ad></ad>
1C18:00 9D 4H B3 A9 01 9D 49 <a6></a6>	1EF0:00 1D AD FC 02 A0 00 C9<0C>	21C8: A4 DB 60 85 D6 85 D7 A0<77>
1C20: 03 A9 05 20 C6 EA 60 20(50)		21D0:08 16 E6 36 E5 26 D6 90<87>
AUROLUG PIT UG AB GO CH ON AUNGUT	TELESCON NO. 12 A2 ME DO 20 EACGES	
LODGE AS EL SE TO PE DO TA PLANCE	1EFB: CØ BØ 12 A2 ØE DD 2C F6<9E>	
1C28:44 F1 4C 72 F1 28 34 F1<70>	1FB0:F0 0B CA 10 F8 20 81 EF<63>	2108:0B 18 B5 E6 65 D7 95 E6<50>
1030:20 65 22 80 88 A4 F2 B1(BA)	1F88:F0 08 CA 10 F8 20 81 EF<63> 1F98:80 C0 05 A0 01 A9 C0 A2<25>	2108:08 18 85 E6 65 D7 95 E6<50> 21E0:90 02 F6 E5 88 D8 EA 60 <ab></ab>
	1FB0:F0 0B CA 10 F8 20 81 EF<63>	2108:0B 18 B5 E6 65 D7 95 E6<50>

Listing 2. Turbo-Basic-Compiler-Runtime-Paket (Fortsetzung)



21F0:D0 F6 A5 8C D0 02 A9 01<BD> 24ha-9h aa h8 9h aa h9 9h aak58) 90(63) 00 DB 90 DC 21FB:85 DS A2 16 A9 00 95 D6<00> 2408: DA 9D 90 EØ EB(3B) 2450:00 E8 DØ EB 2290: CA FB AS 05 DD 1.0 85 F7 A5 A5 85 2208:20 ØD **D4** A2 82 28(19) 24EB: 8E 85 98 85 ØE 85(63) 85 PB (45) 2210:0D F7 **A5** D5 06 ΔØ 24F PH: RE R5 91 85 **PIF** 4C 86 FRK SR > 2218:06 DC 26 DB 26 DA 26 D6<43> 24FR: R5 **R9** 18 A5 90 85 88 A5<3E> 85<975 2500:89 AB A5 91 89 69 2220: 70 18 A5 DC: 65 65 8846F) 1.1 EB 90 10 2228:DC DB 85 DB 90(08) 2508:CD 62 89 00 CC(EF) A5 2230:02 E6 DΑ 88 DØ E2 20 F7 DAKC2> 2510:E5 02 98 0.2 DB 29 83 91 (BE) A9(BB) 22 (8: F.7 EE F7 20 na 20(01) 2518:85 ØF 84 98 84 8E 68 70 €9 F7<45> 2520:02 4C 20 **A5** BE C5 91<12> F1 FF 2240: DD FB 20 20 FI F7 20 20 FB 24 £2<20> 2528: 98 96 A5 8E Ç5 90 BØ 25< 2A> 2248:20 EB 65.593 2530: A5 98 E 9 65 90 85. PE CAS 2250:38 Ę6 E 1 18 A5 02 2258:E6 BS EB AS EA 65 E5 SEK DOX 2539:90 04 C6 91 CA @F AB 88(55) A5 98 90 DØ E9(A4) 82 E9 38 A2 02<C7> 2540: Bt 10 3B 2260: EA 90 E6 2268:85 F5 E9 95 E2 CA 10KBC3 2548:80 85 98 85 BE: 80 64 C6< B5 2 E2 AS EB 98 18 C3 FØ 2F CP / DG > 2550:91 Ç6 GIF 68 20 66(18) 2270:F7 30 2278: ER 38 A5 DC 65 EB 85 DOCEDS 2558: FA F#2 04 F0 59 18 11 A9(DE) €7 A9 4C 2D 85 @2<D3> E9 20<02× 2288: AS DR E5 DB 2560:1C 2C 10 BØ 75<68> F9 F6 FØ 2288:C6 DA 18 A2 02 B5 E 2 2568:66 FA 80 10 F2<4F> 2570:A0 02 E2 CA 10 F 7 30 9E<5A> 68 81 90 48 8847E> 2290: DA 95 68 55 2298: 18 AS 88 65 EØ 85 A5(BE) 579: B1 98 48 68 A2 81 2C A2(39) ØØ(13> 2580:FF A9 **AZ** 63 3A FA AB 20 (0B) 22A0:89 69 88 85 56 60 26 38<20> 69(FØ) 55 91 22A8:86 86 86 56 68 2588: 6A 88 CB 68 AA 18 2280: AS 88 E5 EØ 85 35 A5 89(4F) 2590:03 91 BB CB AB. 48 69 88< 4D > A5 BA(96) 88 84(27) 85 18 2598:91 BC 2298:E9 56 68 BA 68 85 80 48 54 A5 2200:65 EL 85 BB 69 00(F6) 25A0: BD 68 20 40 FØ A9 03 20 52> 220B: F0 ØE 68 38 A5 SIG E5 E1(11) 25AB: 3A FA AØ 88 09 FF 91 88 F 42 91 F2(12) 2200:85 54 A5 22 DØ BB E9 2580: C8 A8 18 49 RES 88 C8<850 2208:A4 C8 AZ 28 B1 21 CBKB@2 69 91 D8 00< 0F 60 25BB: 69 88 88 6C 84 8A(82) 14(91) 2260:80 F8 4C 38 E9 25C0: A6 14 A5 E7 **A4** 12 E4 22EB: 20 DZ EA 28 DD C.3 A6 BA(BZ) 25C8: D0 F6 86 D3 28 1.3 **C18** A9(4B) 91KAE> 2508:89 A& 22F0:86 8A A6 90 88 **A6** FB 20 86 20 ØB. FZ. BB(.38> 28 80 **ac** DØ(82) **A5** 40 08(27) 22F 8: 86 89 66 FA 25DB: C2 D3 20 20 13 2300:00 OB ec. **B1** 98 C5 BA DØ< 7E > 2580:4C D1 CØ 20 02 FB 49 BE C7CO 85 A5432> 25EB: AØ FB 2300:F1 F 47 88 A5 88 90 20 14 F2 AØ (36) A5(B9) 85 FA<86> 31Ø: 89 91 49 10 20 38 25F 0: D4 C9 48 00 82 84 84<A7> D5 231B:C6 A6 92 AØ ch2 BD 00(4A) 25F8:F2 Α9 24 FB 9 68(15) 92 20 51 91(02) 2570: DE 91 SB CS BD 24 DE 2600: 20 51 FR A9 60 20 51 FBsCB> 91 51 54 DE 88 C8/1F > 2608: AB B6 2328:88 CB BD A9 96 4C F 4 48<AA> 91 88 AB CE > C1 2350: BD 7E 0E CB RD 2610:20 EC 20 RØ 20 DD<38> 91(80) 23.39: DE 91 SR CR RD 02 DE 2618:03 68 85 05 28 DR FA ARKARS 91 CB A5-86> 2340:88 CB A5 D4 88 2620:00 A4 D4 CB 40 DØ 25 A4<BB> 2348: D5 91 88 CB **A5** D6 91 88(34) 262B: D5 85 D5 98 AA 40 44 4A(89) C8 79 FB 3A 90 2750:CB A5 **D7** 91 88 45 DR. BR. 7630:4A 20 BA 29 RE 09<28> 2350:91 C8< 7C> 88 CB A5 09 88 2638: 38 09 82 69 256 84 (BB) 91 88 CB A9 88 914 DC> 2640:F2 99 80 41<43> 2368: A5 8A 85 E6 F2 60 2368:88 CB 69 AA 18 69 91(9E) 2648:02 56 86 43 04 32<6E> 05 88 26 2570:88 CB 68 48 69 (2)(2) 91 88 6A 2650:00 20 20 26 84 F3 84 E1(70) 96 98<75> C9 2378:8A 48 A5 DØ 26 A5 2658:84 E2 28 DE 18 BOKEC> FB 68 904BE / 2380:48 A5 91 48 AS 85 2668: 35 85 EØ 20 2D FC 20 DE<1B> 2388: A5 C9 2D 87 85 91 45 D4 48 A6<B6> 2668:FB 30 60 29 26 F8 FB(93) 2390: BA 20 77 EB 68 20 4C F9496 > 2670:20 FC 28 DE F9 C9 3C (42) 69 FB FC(93) 2398:90 07 85 91 84 85 90<8C> 2678:80 10 20 FB 20 ØC. 60 23A0:60 68 68 60 38 49 @D: 55.4 2680:20 06 FC 45 E Ø 64 E.I A6425> 23A8:4E 3D E9 86 8A 20 66 FAK85 > 2688: E2 85 14 84 13 86 12 C5< 79> 90 (05) 2360: BØ F4 D0 F2 AØ 200 P1 2690:14 DØ FA 4C DØ F3 20 DØ< BB> 90KF5/ 23B8:C3 BA DØ AIZ 86 81 2698:F3 4C 9F F3 20 25 CØ E6(FF) 2300:48 85 E (2) CB RA 90 R5 E1475> 2640:F2 80 2648:65 E3 F3 0A 85 E3 ØΔ BACALO 07 85 E2 C8 B1 90< 10 / 85 E3 CØ E6(39) 23CB: CB B1 98 20 23D0:85 CB B1 90 85 CB - 95 4 26B0:F2 E40 E.3 65 €3 60 18 65 (CE) E3 26 90 85 E5 A6 90 20 77KCC 2688:E0 95 EØ 92 Eò E 1 DØ<F4> 23D8: B1 28 F2 26 TTE O: FR 70 EC. F 1 AA BA SICIF CADA: 02 EA 640 PAS. FØ F1(A5) 23E8: EB 4C F9 90 14 68(0B) 2608:26 E2 **A4** A5 EØ 86 E1<0B> 20 £2 68 23F 0: 68 AØ 21F BI 90 48 88 B1(AA) 26D0:06 E.O 26 Εī 26 E2 8 EØK 8E> 26 E1 23F8:90 48 49 10 18 65 98 85<26> 2608:26 E1 E2 65 EØ. 85 FREEAD 91 E6KEF 2400: 90 85 ØE 90 98 E2<11> 84 E6 26E 0: BA 65 85 E 1 65 2408: 0F 68 85 EC AØ 88 B1 90×3F > 26E8:85 E2 68 86 FĊ Α4 E2(C6) 20 2410:05 **D4** DØ ZA CB BI 90 C5<31 · 26F0:A5 EØ A6 E1 86 EØ 26 E1<46> 2418:D5 DB 10 C8 B1 90 C5 D6<16> 26FB: 26 F2 65 FB 85 EØ BA 65(A8) 2420:08 CB B) 90 CS D7 DØ AE> 2700:E1 85 98 65 86 E8<8C> 15 E1 E2 B1 DØ 07 ( CE > 2428: **3**E CB 90 C5 08 2708:26 EI 2A 85 E2 60 20 75(83) 2430: CB B1 90 C5 D9 FØ 86 6A: E3> 2710:FC 86 03 38 E.9 01 CB 88KA63 D4<93> 2718:00 C5 **D**3 80 25 ØA A8<38> 2438:45 EC 45 D4 AB 60 85 29 2440:45 EC 10 F 9 64 49 80 0A< 2F / 2720:68 85 **D4** 85 05 CB B1<10> 68 2448:60 AZ 2728:04 4778:60 191 85 F9 99 FA 70 MAKERS 85 D6 C8 D4 07(95) 20 49 Dt CAK D5> 20 EF \ D2 > D6 88 86 20 2450: A0 20 44 EE 28 69 FB EF C94E2 2738:80 07 A5 D4 64 D5 A6 D3(29) 2458:09 DØ 6C 69 20 2740:60 2748:68 68 68 65 BD APLAED 06 D3 68 98 D3<BB> 2460:C2 00 65 85 85 98 244B: 20 SD AF 22 09 ED RD 79 (BA) 68 69 0.91 48 48 AB(87) E9 BD 71 FC 98 84 69(E7) 2750:28 75 18 20 75 ( BE > 2470:22 22 A9 28 38 09 86 2478: EF 85 80 28 69 EF 85 81 (DE) 2758:FC **D**3 3B E9 01 66 69 92 69 2480:20 EF 85 20 EF (BD) :740: CO 0.0 D/B DIE C5 03 90 DAC 150 3A 2768: AA 85 84 20(41) 03 FA API BB(53> 2488:85 85 20 69 EF A9 20 85 85 91 2490:69 EF A9 6F AD 22 (2A) 2778:A9 FF 28 60 88 AR 65 88 84 87 38 **A5** 84 F9474> 2778:00 85 **D4** 93 91 88(11) 2498:85 7780:C8 91 24001 6F AA A5 B5 E9 22 AB. BASSW 68 85 D5 69 99 8B(7E) 94 AZ FF 86 2788:8A ØA AB 4C FC **A5** D7<23> 2448: 20 48 EF 11<05> 68 88.493 AB 2480:86 EB 86 96 86 86 (BE.) 2790:04 85 DB **B4** DA 2488: FB 66 9B 20 BF FA 20 AAKEE > 2798:84 DC 84 DD AØ 1.0 46 09<90> DC 8B C9 4C 22 (AD) 90 65 EAD 24C0: EA A9 85 6F 27A8:66 98 18 A5 ØF 27A8: DA 85 DC A5 M 65 DB 85<92> 24C8: A9 15 3D E9 A9 00 AAKIB

2780: DD 30 0E 2780: D0 E4 A5 D6 D6 26 DR 68<40> E4 A5 DD 09 16 01KEA> BØ. :7C0:60 FØ 03 3E(21) Δ9 D3 p3<pa>> 2708:€8 98 AB A6 日白 2700:18 69 23.1 **B5** DH 48 98 69(32) 30 48 92<302 2708:00 £6 B5 D9 C9 69 27EB: 84 92 18 B9 98 DE @1<7E) 69 2758:85 D6 20 DE 00 30<FØ2 27FØ:00 85 07 20 00 FC PLA. DCK ZAD 2768:26 DC 26(A9) DD **A4** DD A5 OA : 60 : 60H 65 DE 83 DC 98 45 DD: 900 DB< 70> 2888:85 DD 68 85 09 68 85 25.1 Dt 40 6C F D 03 85 DA RAKAD) 86 2818:09 85 FØ< 96> DC 84 DD 85 09 98 69 20 85 D6 (5A) 1920: A0 90 310 BD 00 D9 2928:85 07 DZ. DOK BB3 A5 18 10: 90 8E 85 D4 A5 SE 85<@F> DF 20 29 78: D5 AS DE 84 DD EC(63) 89 AB 472 240: A6 **D3** EB 20 20 51 FØ ØA 2848:46 DD 91 **D4** 68 DØKFD> 850:FB E6 D5 CA D/B F6 AA DEK SCO CA 2858:FØ Ø6 91 **D4** CS DØ FAC9B) 84<1E 84 E7 AB 96 84 2460:60 E8 2868: E9 68 10 85 EB 84 C6<587 92 8/0:92 A4 **R9** 24 DE 85 E7(29) DE 28 / B: 89 €6 65 77 20 20 EB(C2) A5 E9 "880: A5 EB DB E5 D9<A2> 06 35 C4 A5 2888:00 44 A5 E7(80) E6 D7 20 D4(9E) 1890: F5 P471 2R E7 18 A5 DC 65 EB 85 DCKE9> 2898:FC E9 BA0: A5 DD 65 06 DC 24 B5<4E> 28A8: DD AB A5 DC BA 26 DĎ 65(A7) 98 2880: DC AA 65 DD 65 AB(EØ) D590 CB 7888:8A 65 04 01 60 40<500 .908:81 FB 20 63 FD AC. 85 EB< 762 2868: 20 AD FD 4C B5 EB 86 D3(E7) BD0: 29 A6 D3 13 CØ 86 03 20<BD> 28D8:F5 FR 34 EA A5 **A5** D4<302 28E 0: A4 05 A6 03 20 A.S. FD 40<2A> EC 28L8: 5F 88 D3 20 13 CØ A6<55> 11(11) 2HF 0: D3 96 D3 20 EA AS 78F8:F8 18 A5 D4 O.A. D5 **A6** D3<DØ> 790:20 AD FD 40 5F EC 28 Ø4<D0> 2908:FE 4C CE ED 20 ØA 40<66> FE 2910:CE ED A9 82 4C ae E9 86<60> 291B: B3 AA. 1.1 FA E5. 86 D4 BAK DA > A6 D5 89 86 2920:BB 86 D6 86(A7) 2928:80 A6 107 BD 86 A6 D3 4C<1B> 2930:77 EB 20 AF 59 FF A5 D9 A4(5A) 2938: D8 FE 9D 99 98<E7> 20 DB 20<31> 2940:9D 92 DA 60 20 59 294R-01 FF 20 AR FF 4C 7E FECAS: 2950:20 59 FE 20 6.0 FE 28 ABS &C > 2958:FE DD 22 DB 98 BA DØ DOCKBE 2960: 4B 98 DD PACA. DA 68 Eq2 04540 296B:68 A5 D.7 A4 D6 CS BD 90 90> 85KDF 2970:06 DØ 08 C4 80 80 84 2978:8D 84 80 18 45 04 85 2980:65 SEC 85 EØ 45 DS. 85 BBKES> 2988:65 BD 20 26 22 85 E1 38(20) 2990: A6 03 A5 E 10 FD 00 DB A8 . SC 2998:A5 E1 FD 88 D9 69 92(76) C6 92 29AR: AA HD 99 DE BS. F2 RD(87) 29AB: 2A DE 95 ĘZ FØ **BA<7F** 85 E.3 2980:A5 D8 CS AS 09 65 E3kCB: 29BB: 80 (AD 49 05 40 3D E9 ASKEQ > 85 2900:08 85 AS D9 E3 CAKBEN F2 2908:92 A6 92 28 BD 20 E9<1D/ DE 29D0:01 85 EB BD ĎE E9 26 00.46 2908:83 E1 90 DE A5 E2 FS. EBK 6B D6 AS. EZ C1 2960:05 E5 85 D7<DC> 2968:90 FØ DØ 05 D6 CC 18 A5 88 29F 01 D4 65 EØ 85 04 A5 DS 65<DD0 29FB: E1 85 85 £2 84 E3<14> 05 60 FØ 2A00:05 E3 28 ΕB A5< 42 86 E5 2A08: D6 C5 E2 **A5** D7 BRK RES E3 2A10:86 90 48 20 EB A6<F72 86 E2 A6 D7 86 68(9A) 2A18:06 E3 2A20:38 F9 91 85 EØ 90 4C 18409 242B: FF 85 D4 49 21 AB 48 BAKSES FB BA ZAJØ: 06 86 07 92< 7B / 16 A6 2A38: A5 D4 48 98 48 CA BD 004 B32 2449 - DE BC. 2A ÐE C6 D6 DØ F2KF00 ZA48:85 04 92 86 84 D5 A5 D7<BD> 2A50:48 4C 17 21 **A5** 03 AØ 99 38 ZA58: 4C 84 13 CØ A5 BA 88 4C<01 2A60:13 CØ 86 **D4** 18 45 D4 90 (CA) 38(F1) 2A6B: 03 CB FØ ØD 60 86 D4 2A70:E5 D4 **B49** 88 CO 2A/B:F3 4C 29 F5 00 00 00 00 00(CD) Laenge 10876 Bytes

Listing 2. Turbo-Basic-Compiler-Runtime-Paket (Schluß)

# Von der Ordnung zum Chaos -Apfelmännchen auf dem Atari

Faszinierende Grafiken von nahezu unendlicher Vielfalt erzeugt das Programm »Apfelmännchen« auf dem Atari 800XL in Turbo-Basic.

rundlage dieses Programms ist ein Teilgebiet der Mathematik von großer Komplexitat. Es handelt sich um die Berechnung und Betrachtung der sogenannten Mandelbrot-Menge und speziell deren Grenzschichten. Diese Menge, benannt nach dem Wissenschaftler Benoit B. Mandelbrot, besteht aus einer Folge von komplexen Zahlen, die sich recht merkwürdig verhalten. Trägt man die beiden Komponenten jeder komplexen Zahl, die sich aus diesem Algorithmus ergibt, auf die Abszisse und die Ordinate eines Koordinatensystems auf, so erhält man Grafiken, die kuriose Eigenschaften aufweisen.

Um den grundlegenden Algorithmus zu verstehen, muß man sich etwas näher mit den Eigenschaften komplexer Zahlen auseinandersetzen. Eine komplexe Zahl besteht immer aus zwei Komponenten, dem Realteil und dem Imaginärteil. Die Schreibweise erfolgt in der Form: »C=a+bi«. Die komplexe Zahl C ergibt sich also aus dem Realteil a und dem Imaginärteil b. Der Buchstabe i steht für eine Konstante und unterliegt folgender Definition: »i²=-1«. Die Rechenregeln der komplexen Zahlen weichen jedoch von denen der reellen Zahlen ab. Geht man bei der Addition noch in gewohnter Weise vor und addiert Real- und Imaginärteil einfach separat, ist das Multiplizieren schon etwas komplizierter. So ergibt sich »(a+bi)\*(c+di)« zu »(a\*c-b\*d)+(a\*d+b\*c)\*i«. Wer diese Formel genauer betrachtet, wird feststellen, daß sich »i²« aufgrund seiner Definition zu -1 aufgelöst hat.

Beim »Apfelmännchen« geht man von der komplexen Zahl Null aus, zieht eine komplexe Zahl C ab und quadriert das Ergebnis. Nun zieht man erneut Cab und quadriert das Ganze wiederum. Dieser Vorgang wiederholt sich so lange, bis ein vorher festgelegtes Grenzkriterium überschritten wird. Alle Zahlen, die hierbei gegen Unendlich streben, gehören zur Mandelbrot-Menge, Am Rand dieser Menge stellt sich jedoch ein seltsamer Effekt ein. Hier fallen oder steigen die Zahlen bei wechselndem Vorzeichen scheinbar völlig regellos. Dies ist der für die Grafik interessante Teil der Mandelbrot-Menge. Zwei Grenzkriterien legen nun fest, ob man sich innerhalb oder außerhalb beziehungsweise am Rand der Mandelbrot-Ebene befindet. Das erste Kriterium bricht den Algorithmus nach einer vorher festgelegten Zahl von Durchgängen ab. Im Listing geschieht dies durch die Variable »TMAX«. Wird dieser Wert erreicht, nimmt man an, daß die Folge gegen Unendlich strebt. Der Punkt auf dem Bildschirm erhält die Farbe Schwarz. Im anderen Fall wurde das zweite Grenzkriterium zuerst erreicht. Dies ist dann der Falt, wenn die komplexe Zahl einen bestimmten Wert überschreitet. Im Listing wurde der Grenzwert 8 angenommen, der jedoch keine feste Größe darstellt, und fast beliebig geändert werden kann. Die zuzuordnende Farbe wird nun von der Anzahl der Iterationen abhängig gemacht. Im Listing geschieht dies mit der Zuordnung »TIEFE MOD 3+1«. Diese Berechnung kann zu den Farben Eins, Zwei und Drei, niemals jedoch zu der Farbe Null führen.

Für jeden Punkt, der berechnet werden soll, wird natürlich eine andere komplexe Zahl C gewählt. Die Umgebung der Mandelbrot-Ebene ergibt sich für folgende Werte: Realteil von -2 bis 1 und Imaginarteil von -1.5 bis 1.5. Innerhalb dieser Grenzen kann man nun beliebig kleine Ausschnitte auf dem Bildschirm darstellen, also Teilbereiche der Menge gewissermaßen vergrößern. Dabei stellt man fest, daß immer neue Grafiken oder Figuren entstehen, die zum Teil zu anderen Ausschnitten ähnlich, aber nie genau gleich sind. Die einzige Einschränkung, die hierbei vom Computer auferlegt wird, liegt in der begrenzten Rechengenauigkeit. Bei theoretisch unendlicher Rechengenauigkeit ließe sich die Grafik unendlich oft vergrößern. Geht man also beispielsweise von den Grenzen -1,1,-1 und 1 aus, so wären als nächste Vergrö-Berungsstufe die Werte -0.5,0.5,-0.5 und 0,5 denkbar. Halbiert man diese Werte immer wieder, so ist klar ersichtlich, daß auch eine zehnstellige Rechengenauigkeit bald nicht mehr zur Darstellung der Zahlen genügt.

### Komplexe Formel, schöne Grafik

Zur Realisierung dieses Programms auf dem Atari wurde als Programmiersprache Turbo-Basic XL. gewahlt. Die zirka viermal so hohe Rechengeschwindigkeit, die diese Sprache gegenüber dem normalen Basic aufweist, ist für eine akzeptable Rechendauer unbedingt notwendig. Die Grafik wird in Grafikstufe 15 erzeugt. Dies bedeutet, daß genau 30720 Bildpunkte berechnet werden müssen. Abhängig von der maximalen Iterationszahl, die man mit \*TMAX« bestimmt hat, kann jeder Punkt zudem bis zu 250mal berechnet werden. Selbst unter Turbo-Basic XL führt dies in Extremfällen zu Rechenzeiten von acht Stunden und mehr. Die Programmierung in Atari-Basic erscheint also wenig sinnvoll, will man nicht ein bis zwei Tage auf die Vollendung einer Grafik warten.

#### Handhabung des Programms

Nach dem Starten des Programms kommt man in das zentrale Menü, von dem aus alle Funktionen direkt ansprechbar sind. Punkt eins des Menüs betrifft den Start eines Apfelmännchens. Man muß zuerst die obere und untere Grenze des Real- und Imaginärbereichs eingeben. Auf die Frage

PROGRAMM-STECKBRIEF				
Programmname	Aptelmännchen			
Programmtyp	Grafik			
Programmlersprache	Turbo-Basic XI.			
Programmlänge	7155 Byte			
für Computer	800 XL/130 XE			
zusätzliche Hardware	Diskettenlaufwerk			
Eingabehilfe	Prüfsummer			
Bemerkung	Erzeugt und speichert Grafiken, die mittels einer speziellen mathematischer Formel berechnet werden. Auf der Leserservice-Diskette sind emige fer- tige Grafiken bereits gespeichert.			
Leserservice	Diskette (APFELTUR)			

»TIEFE?« antwortet man mit der maximalen Anzahl der Iterationen, die man zur Berechnung jedes Punktes zulassen will. Je höher man diesen Wert wählt, desto feiner werden die Grenzbereiche zur Mandelbrot-Ebene berechnet. Dies bedeutet allerdings auch eine Erhöhung der Rechenzeit, so daß man hier meist Kompromisse eingehen wird. Die Berechnung der Grafik kann dann auf dem Bildschirm mitverfolgt werden. Nach der Fertigstellung springt das Programm zurück ins Hauptmenü. Das fertige Bild kann man nun mit Option 2 auf dem Bildschirm betrachten, Mit den Tasten 0, 1, 2, und 3 werden die vier Grundfarben nach Belieben nachträglich geändert. Mit Hilfe des dritten Menupunktes speichert man das fertige Bild. Wie es auch im Micropainterformat üblich ist, werden am Ende des Bildes vier Byte angehängt, die Auskunft über die verwendeten Farben geben. Zusätzlich speichert das Programm allerdings noch die eingegebenen Parameter, So kann man mit Punkt vier des Menüs also auch nur solche Bilder von Diskette laden, die das eben beschriebene Format besitzen.

### Apfelmännchen schön und bunt

Das Inhaltsverzeichnis einer Diskette erhält man mit Punkt fünf des Menüs. Der sechste Punkt, nämlich »Effekte«, läßt die drei Vordergundfarben eines Apfelmännchens über den Bildschirm »fließen«. Dies gibt der Grafik je nach Aussehen interessante, teilweise räumlich wirkende Effekte.

Um das Vergrößern einzelner Bildteile zu erleichtern, wurde Menüpunkt sieben aufgenommen. Mit Hilfe eines Joysticks kann man einen Bildausschnitt eingrenzen und erneut berechnen lassen. Dies geschieht, indem man zwei Eckpunkte eines gedachten Rechtecks entsprechend auf dem Bildschirm plaziert. Welchen der Eckpunkte man bewegen will, wird mit der Feuertaste gewählt. Die Berechnung der neuen Grafik beginnt nach dem Drücken der START-Taste. Will man sich die neuen Parameter vorher noch ansehen, so wählt man einfach OPTION. Funktion acht des Menüs zeigt nun die Parameter, und mit Punkt neun startet die Berechnung des neuen Apfelmännchens.

#### Zum Programmaufbau

Das Programm »Apfelmännchen« wurde in Turbo-Basic XL geschrieben und konnte daher nicht nur wesentlich schneller, sondern auch strukturierter als im normalen Basic angelegt werden. So wird es auf den ersten Blick auffallen, daß beispielsweise auf Zeilennummem-orientierte Sprunganweisungen mit Ausnahme der »TRAP«-Befehle vollständig verzichtet wurde. Alle Hauptprogrammteile und Unterroutinen sind als Prozeduren angelegt und sinngemäß benannt.

Die Hauptschleife besteht aus einer DO-LOOP-Anweisung von Zeile 95 bis Zeile 240. Hier wird das Hauptmenü auf den Bildschirm gebracht, und von hier aus erfolgt der Sprung zu allen wesentlichen Prozeduren. Die einzelnen Prozeduren:

#### DECLARE

Es erfolgt die Dimensionierung aller Stringvariablen sowie die Erstbelegung der Variablen für die Änderung der Farbregister. Das Aussehen der beiden Player, die für die Ausschnittberechnung benötigt werden, legen die Variablen LO\$ und RU\$ fest.

#### RECHNEN

Hier findet man den Berechnungsalgorithmus, wie er im wesentlichen bereits erklärt wurde. Verwendung findet die Grafikstufe 15; man verfügt also über vier Farben bei einer Auflösung von 160 mal 192 Bildpunkten. Es wäre auch eine andere Grafikstufe denkbar. In diesem Fall müßten folgende Änderungen vorgenommen werden: Anpassung der Variablen DX, DY sowie der Schleifen ZEILE und SPALTE an die

neue Auflösung. In Zeile 1230 muß anstatt der drei die Anzahl der Farben minus eins eingesetzt werden.

#### **EFFEKTE**

Die drei Farbregister für die Vordergrundfarben werden mit verschiedenen Farbwerten belegt. Durch die Verschiebung der Farben entsteht der Eindruck des Fließens.

#### PARAMETER

Hier gibt man die Randwerte für ein neues Apfelmännchen ein. Ist dies geschehen, wird die Prozedur RECHNEN aufgerufen.

#### SEHEN

Über die Prozedur BILDHOLEN wird das momentan im Speicher befindliche Apfelmännchen auf dem Bildschirm dargestellt. Die Farbregister 708, 709, 710 und 712 werden durch Drücken der Tasten 0 bis 3 jeweils um den Wert eins inkrementiert. Die REPEAT-Anweisung wird bis zum Betätigen der Leertaste ausgeführt

#### **SPEICHERN**

Zuerst erfolgt der Aufruf der Prozedur »FILEGET«. Dabei wird der Filename des Bildes gelesen, auf Fehler überprüft und gegebenenfalls ergänzt. Mit dem BPUT-Befehl wird der Inhalt des Bildschirmspeichers auf Diskette abgelegt. Es folgen die vier Farbbytes und die Parameter. Da es sich bei letzteren um Gleitkommazahlen handelt, werden sie als Strings gespeichert.

#### LADEN

Diese Prozedur entspricht fast vollständig der Prozedur »SPEICHERN«. Alle Schreibbefehle sind sinngemäß in Lesebefehle umgewandelt. Eventuell auftretende Fehler bei der Diskettenhandhabung werden von TRAP-Routinen abgefangen.

#### DIRECTORY

Der linke Bildrand wird auf Position 7 gesetzt und das Directory mit dem DIR-Befehl aufgerufen. Nach Betätigung einer Taste erfolgt der Rücksprung ins Hauptmenü.

#### **Ausschnitt**

Diese Prozedur setzt sich aus mehreren Unterprogrammen zusammen. PMINIT initialisiert die grundsätzlichen Voreinstellungen für den Player/Missile-Aufruf. Verwendung finden die Player null und eins. Sie werden in Einzelzeilen-Auflösung dargestellt. Ihre Farbe ist Weiß und die erste Position wird durch die Varlablen XO, YO, X1 und Y1 definiert, PMBUILD positioniert die beiden Player auf dem Bildschirm. Der Aufbau erfolgt dabei durch den MOVE-Befehl. Die folgende REPEAT-Anwelsung fragt den Joystick und die Sondertasten ab. Abhängig von der Auslenkrichtung des Joysticks werden die »PMLEFT«, »PMRIGHT«, »PMUP« Prozeduren »PMDOWN« angesprungen. Sie versetzen den jeweiligen Player um einen Pixel in die gewünschte Richtung. Ist die Positionierung abgeschlossen, werden die alten Parameter durch die neuen Werte ersetzt. Die Prozedur »PMEXIT« schaltet zuletzt die beiden Player wieder aus.

#### PARAMANZ

Hier werden lediglich die aktuellen Parameter auf dem Bildschirm angezeigt. Nach dem Betätigen einer beliebigen Taste gelangt man wieder ins Hauptmenü.

(Wolfgang Czerny/wb)



10	<yw>  </yw>	1410 POKE 710,COL0:PAUSE 5	<f< th=""></f<>
15 REM - Apfelmaennchen in -	T		
17 REM - Turbo-Basic XL fuer -	<fu></fu>	1420 POKE 708, COL2: POKE 709, COL0	⟨P
		1430 POKE 710, COL1: PAUSE 5	<6
19 REM - Atari-Computer mit -	(CE)	1455 UNTIL PEEK (764) <> 255	≺X
	<bk></bk>	1460 ENDPROC	<p< td=""></p<>
5 REM -	<km></km>	1470	< M
	<ra></ra>	1480 REM - Parametereingabe -	<m< td=""></m<>
	<ar></ar>	1490	< W
0 REM - (c) Happy-Computer -	<ll></ll>	2000 PROC PARAMETER	≺U
5 —	<zi></zi>	2005 REPEAT	<e< td=""></e<>
Ø EXEC DECLARE	<yx></yx>	2010 TRAP 2010	<l< td=""></l<>
5 <b>00</b>	<ra></ra>	2020 CLS :POKE 752,0	≺N
00 GRAPHICS 0: POKE 752.1	<gu></gu>	2030 POSITION 9.3:? "PARAMETER_EINGEBEN"	<w< td=""></w<>
10 POKE 709,0:POKE 710,10:POKE 712,10		2040 POSITION 9,4:? "	
15 POKE 82.0	<hv></hv>	2045 REPEAT	ΚĒ
,	<xc></xc>	2050 POSITION 3,7: INPUT "Linker_Rand_=_"	4.1
	T	· · · · · · · · · · · · · · · · · · ·	<8
30 POSITION 9,2:? ""	<bc></bc>	; XMIN	10
40 POSITION 3,5:7 "(1) ALL ParameterLeing		2060 POSITION 3,7:INPUT "Rachter_Rand_=_	- 5 1
ben"	<sm></sm>	"; XMAX	<h< td=""></h<>
50 POSITION 3,7:7 "(2)Bild_ansehen"	<aw></aw>	2065 UNTIL XMIN <xmax< td=""><td><p< td=""></p<></td></xmax<>	<p< td=""></p<>
60 POSITION 3,9:? "(3)Bild_speichern		2067 REPEAT	<f< td=""></f<>
	<dv></dv>	2070 POSITION 3,11:INPUT "UntereraRanda=	
70 POSITION 3,11:7 "(4)Bild_laden"	<mg></mg>	_" ₹YMAX	<5
90 POSITION 3,13:? "(5)Directory"	<us></us>	2080 POSITION 3,13: INPUT "Oberer_Rand_=_	
90 POSITION 3,15:? "(6)Effekte"	<gc></gc>	":YMIN	< W
75 POSITION 3,17:? "(7) AAAAuschnitt"	<jf></jf>	2082 UNTIL YMAXKYMIN	<p< td=""></p<>
77 POSITION 3,17:? "(7) ALAHUSERNITT"		2083 REPEAT	₹F
			41
igen"	<wk></wk>	2085 POSITION 3,15: INPUT "Tiefe_(5_~_254	
99 POSITION 3,21:? "(9)Rechnen"	<rf></rf>	) _="; TMAX	< 0
20 POSITION 3,23:? "(0)Ende":	<ea></ea>	2087 UNTIL TMAX>4 AND TMAX<255	<₩
05 REPEAT	<dm></dm>	2090 POSITION 3,20:? "ALLESAINAORDNUNGA?	
18 GET EIN	<he></he>	_(J/N)";	< 0
20 UNTIL EIN>47 AND EIN<58	<0H>	2091 REPEAT	KF
25 IF EIN=48 THEN EXIT	KEF>	2092 GET EIN	<3
30 ON EIN-48 EXEC PARAMETER, SEHEN, SPEIC		2093 UNTIL EIN=74 OR EIN=78	<z< td=""></z<>
ERN, LADEN, DIRECTORY, EFFEKTE, AUSSCHNITT,		2074 UNTIL EIN=74	₹K
			<k< td=""></k<>
ARAMANZ, RECHNEN	<xd></xd>		
40 LOOP	(CF)	2100 ENDPROC	<0
50 END	<xx></xx>		< 4
300	<ve></ve>	2120 REM - Bild ansehen -	<d< td=""></d<>
002 REM - Rechenroutine -	<iu></iu>	2138	< 9
002	<vq></vq>	3000 PROC SEHEN	< Y
809 PROC RECHNEN	<nr></nr>	3010 EXEC BILDHOLEN	< I
010 GRAPHICS 15+16:POKE 764,255 '	<@S>	3015 REPEAT	<e< td=""></e<>
	<mn></mn>	3017 REPEAT	<f< td=""></f<>
	(KE)	3020 GET EIN	< I
	<b>KSU</b> >	3048 UNTIL (EIN>47 AND EIN<52) OR EIN=32	
080 FOR ZEILE=0 TO 191	(FH)	3050 IF EIN-48=0 THEN COL0=COL0+1: IF COL	_
	<mh></mh>	0>255 THEN COL0=COL0-256	<z< td=""></z<>
292 FOR SPALTE=0 TO 159			~ ~
110 TI=0:XW=0:YW=0:X2=0:Y2=0	<jy></jy>	3060 IF EIN-48=1 THEN COL1=COL1+1: IF COL	20
IZE WHILE TIKTMAX AND (X2+Y2) (8	<py></py>	1>255 THEN COL1=COL1-256	< 1
120 WHILE TIKTMAX AND (X2+Y2)<8 130 YW=2+XW+YW-CY:XW=X2-Y2-CX 150 X2=XW-2:Y2=YW-2:TI=TI+1	<tu></tu>	3070 IF EIN-48=2 THEN COL2=COL2+1:IF COL	
150 X2=XW^2:Y2=YW^2:TI=TI+1	<xn></xn>	2>255 THEN COL2=COL2-256	<r< td=""></r<>
180 WEND	<ue></ue>	3080 IF EIN-48=3 THEN COL3=COL3+1: IF COL	
190 IF TI=TMAX	<fi></fi>	3>255 THEN COL3=COL3-256	<a< td=""></a<>
	<ql></ql>	3090 POKE 708, COL0: POKE 709, COL1: POKE 71	
220 ELSE	<te></te>	0,COL2:POKE 712,COL3	< X
	<id></id>	3100 UNTIL EIN=37	₹E
	<5W>	3100 UNTIL EIN=32 3110 EXEC BILDSPEICHERN	<l< td=""></l<>
DEG DY-DY-DY-DY-OT COALTE TEXTE		TORREST DILLOSI CIONERIN	
250 CX=CX+DX:PLOT SPALTE, ZEILE		3900 ENDPROC	<0
260 IF PEEK (764) = 28 THEN POP : GOTO 1295		3990 3992 REM - Bild auf Diskette -	<
230 CX=CX+DX;PCOT SPALTE,ZEILE 260 IF PEEK(764)=28 THEN POP : GOTD 1295 270 NEXT SPALTE 280 CX=XMIN:CY=CY-DY 290 NEXT ZEILE EXEC BILDSPEICHERN	<db></db>	SYYZ REM - Bild auf Diskette -	< K
2BO CX=XMIN: CY=CY-DY	<dc></dc>	3993 REM - speichern -	< 6
290 NEXT ZEILE	<gb></gb>	3994 —	< X
295 EXEC BILDSPEICHERN	<wl></wl>	4000 PROC SPEICHERN	<5
300 ENDPROC	<01>	4010 CLS	<t< td=""></t<>
301	<v0></v0>	4020 POSITION 9,3:? "BILDASPEICHERN"	< E
302 REM - Bildeffekte -	<dv></dv>	4030 POSITION 9,4:? "	⟨T
TOT	- 1	AREA EVEN EN POST	
303	<\M>	4050 EXEC FILEGET	<i< td=""></i<>
305 PROC EFFEKTE 318 EXEC BILDHOLEN 370 POKE 764,255	<jt></jt>	4100 EXEC BILDHOLEN	< 1
318 EXEC BILDHOLEN	<if></if>	4110 OPEN #1,8,0,DAT\$	<5
370 POKE 764,255	<gs></gs>	4120 BPUT #1,DPEEK(88),7680	<e< td=""></e<>
3/5 REPEAT	<fu></fu>	4130 PUT #1.COL0.COL1.COL2.COL3	<p< td=""></p<>
		4140 PRINT #1.STR\$ (XMIN)	< D
	<hi></hi>	4000 EXEC FILEGE: 4100 EXEC BILDHOLEN 4110 DPEN #1,8,0,DAT* 4120 BPUT #1,DPEK(88),7680 4130 PUT #1,COL0,COL1,COL2,COL3 4140 PRINT #1,STR*(XMIN) 4150 PRINT #1,STR*(XMAX)	₹F
	41147	TAME TIME TANGETH TANK TO THE TANK TO THE TANK T	41
·	<rg></rg>	4160 PRINT #1,STR\$(YMIN)	ΚE

Turbo-Basic XL-Listing zu »Apfelmännchen«

•	<60>		XDELTA=X1POS-X0POS	<0
180 PRINT #1,STR\$(TMAX)	<cv></cv>		DSQUND	< 3
190 ELOSE #1	<np></np>		UNTIL (START-6 OR START 5) AND XDEL	
900 ENDFROC	<0X>		AND YDELTA>1	<1
990	<xa></xa>		DEX-203-40: DEY=223-32	<1
992 REM - Bild von Diskette -	<@J>		DXD=(XMAX-XMIN)/DEX	< (
993 REM - laden -	<zh></zh>	8180	DYD= (YMAX-YMIN) /DEY	<₹
994	<xd></xd>		XMINN=XMIN+DXD+(X@PQS-44)	<1
000 PROC LADEN	<rm></rm>		XMAXN=XMIN+DXD*(X1POS=35)	< 6
010 CLS	<tm></tm>		YMINN=YMIN+DYD+(Y0POS 30)	< F
020 POSITION 9,3:2 "BILD_LADEN"	<16>	_	YMAXN YMIN+DYD*(Y1POS-28)	ζį.
DEG PUSTITUR 7,517 "BILDACHDEN"				
030 POSITION 9,4:? ""	<01>		TMAX=254	<1
050 EXEC FILEGET	<10>		XMIN=XMINN: XMAX=XMAXN	<)
100 GRAPHICS 15+16	<fr></fr>	8250	YMIN=YMINN: YMAX=YMAXN	<6
110 OPEN #1.4.0.DAT\$	<qs></qs>	8260	EXEC PMEXIT	<1
110 OPEN #1.4.0,DAT\$ 120 BGET #1,DPEEK(S8),7680	<xh></xh>	8270	IF START=6 THEN EXEC RECHNEN	<7
125 GET #1,COLØ,COL1,COL2,COL3	<jp></jp>	8900	ENDPROC	< F
130 INPUT #1.Es:XMIN=VAL(Es)	<6M>	8990		< >
			REM - Player Missile-Grafik -	<3
140 INPUT #1,Es: XMAX=VAL (E\$)	<in></in>			
150 INFUT #1,E\$:YMIN=VAL(E\$)	<hk></hk>		REM - initialisieren -	<0
168 INPUT #1,Es;YMAX=VAL(Es)	<2F>	8793		<>>
165 INPUT #1,E\$: TMAX=VAL(E\$)	<6T>	9000	PROC PHINIT	< E
170 CLOSE #1	<nk></nk>	9010	PMB=PEEK (106)-40: POKE 54279, PMB	<1
180 EXEC BILDSPEICHERN	< VQ>		PMB=PMB#256: PDKE 559,62	< >
190 ENDPROC	<pj></pj>		POKE 53256.0: POKE 53257.0	<1
199 —	<xv></xv>		PMNR=8: POKE 704,255: POKE 705,255	Č.
200 REM - Directory auf dem -	<pr></pr>		MOVE ADR (LEER#), PMB+1024, 256	< E
210 REM - Bildschirm ausgeben -	<p1></p1>		MOVE ADR (LEER\$), PMB+1280,256	<f< td=""></f<>
220	<vs></vs>		X0P05-44: Y0P05: 30	<1
000 PROC DIRECTORY	<76>	9080	X1POS-198:Y1POS=218	< 6
010 CLS : POKE 82,7:7	<x6></x6>	7070	ENDPROC	<f< td=""></f<>
020 DIR	<st></st>	9892	_	<1
030 GET EIN	<in></in>	7073	REM - Player Missiles auf	< F
	<lp></lp>		REM - bauen und mani	<1
035 POKE 82,0			REM - pulieren -	<€
840 ENDPROC	<bt></bt>	9073	7	<
250	<vy></vy>			
060 REM - Filename abfragen -	<vx></vx>		PROC PMBUILD	<1
070 —-	<we></we>		POKE 53277,3	<€
000 PROC FILEGET	<mb></mb>	9120	POKE 53248, X0POS: POKE 53249, X1POS	-≪F
005 TRAP 7005	<rc></rc>	9130	MOVE ADR(LO\$),FMB+1024+Y0F05,8	<1
207 POKE 752,0	<pg></pg>		MOVE ADR (RUS) .PMB+1280+Y1POS.8	<6
010 POSITION 3.7: INPUT "WELCHER_FILE-N	- 1		ENDPROC	< 5
to Branchia	TI >		PROC PMEXIT	< 8
E_";FILE\$	⟨JL⟩			<1
WZW DAIS="AAAAAAAAAAAAA	<80>		POKE 53277,8	
050 IF FILE#(1,2)<>"D:"	<en></en>		POKE 559,34	5
Pars   Piles   Pars   P	<tb></tb>		MOVE ADR (LEER\$) ,PMB+1024,256	< E
070 DAT\$(1,2)="D:"	<ch></ch>	9240	MOVE ADR (LEER\$), PM8+1280, 256	<f< td=""></f<>
080 ELSE	<ty></ty>	9250	ENDPROC	< F
090 DATS=FILES	<ud></ud>	9290	_	< 6
095 ENDIF	<uh></uh>	9292	REM - Player Missiles nach -	<1>
	<nk></nk>		REM - oben bewegen -	<1
110 TRAP 20000		7294	_	<
900 ENDPROC	<pa></pa>			-⟨ί
990	<xd></xd>		PROC PHUP	
992 REM - Ausschnitt waehlen	<mb></mb>		IF PMNR=0	<1
993 REM - fuer diesen Programm	<pa></pa>	9312	Y@POS=Y@POS-1	KI
994 REM - teil wird ein Joy	<ya></ya>	9314	IF YOPOS<30 THEN YOPOS=30	< 2
995 REM - stick benoetigt -	<vl></vl>	9317	MOVE ADR(LO#),PM8+1024+Y0POS,8	<(
996	<yb></yb>	9318	ENDIF	<1
000 PROC AUSSCHNITT	<05>		IF PMNR=1	<1
			Y1P0S=Y1P0S-1	<1
818 EXEC BILDHOLEN	<16>		IF YIPOS<28 THEN YIPOS=28	- CI
015 EXEC PMINIT	<zw></zw>			
020 EXEC PMBUILD	<na></na>		MOVE ADR(RU\$),PMB+1280+Y1POS,B	<1
022 REPEAT	<eu></eu>		ENDIF	-KI
024 REPEAT	<fc></fc>	9390	ENDPROC	<
030 JOY=STICK(0)	<cp></cp>	9392	_	< 2
040 PRESS=STRIB(0)	<fg></fg>	9393	REM - Player Missiles mach -	< 8
045 START=PEEK (53279)	<1L>		REM - unten bewegen -	<1
		9399		<1
858 UNTIL JOY<>15 OR PRESS=8 OR START<			PROC PHOOMN	3
	<nq></nq>			
060 IF JOY=11 THEN EXEC PMLEFT	<aa></aa>		IF PMNR=0	<1
070 IF JOY=7 THEN EXEC PMRIGHT	<e0></e0>		Y@P05=Y@P0S+1	<
080 IF JOY=14 THEN EXEC PMUP	<yx></yx>	9414	IF Y0POS>221 THEN Y0POS=221	<.
090 IF JOY=13 THEN EXEC PMDOWN	<ql></ql>	9416	MOVE ADR(LO\$),PMB+1024+Y0POS,8	<1
100 IF PRESS=0 AND PMNR=0	<my></my>		ENDIF	</td
			IF PMNR=1	</td
105 PMNR=1:SOUND 0,10,10:PAUSE 10			Y1POS=Y1POS+1	ζI
107 ELSE	<uf></uf>			
110 IF PRESS=0 AND PMNR=1 THEN PMNR=0:			IF Y1POS>218 THEN Y1POS=218	< E
	<uu></uu>	9426	MOVE ADR(RUS), PMB+1280+Y1PD5,8	< 1
UND 0,100,10,10:PAUSE 10	(00)			
UND 0,100,10,10:PAUSE 19	<5V>	9428	ENDIF	</td
			ENDIF ENDPROC .	<1

# LOGO

Jeder kann programmieren Computersprache für Eltern und Kinder DANIEL WATT

LOGO...Ergebnis der Erforschung menschlicher Intelligenz

Entwickelt von Seymour Papert, Pädagoge und Mathematik professor

Erste Computersprache, die bewußt Strategien menschlichen Denkens dient – und in ihrer Logik der Realität gerecht wird LOGO ersetzt BASIC, sagen Pädagogen und Mathematiker LOGO kommt dem übergreifenden, assoziativen Denken entgegen. BASIC dagegen ist ein Setzkasten von Logik-Buchstaben.

DANIEL WATT... hat im Team von Seymour Papert gearbeitet und ein Buch geschrieben, das voller Bilder seine Erlebnisse mit Kindern am Computer wiedergibt. Ein hochwertiges Textbuch für LOGO-Kurse. Ein Buch für Lehrer, die nach einem bereits von Schulbehörden empfohlenen LOGO-Kursbuch suchen.



"Buch des Jahres 1983" in den USA

12:11

te wi Verlag GmbH Theo-Prose!-Weg 1 8000 Munchen 40

Ein Buch für APPLE II, C-64, IBM PC, ATARI bis 520 ST., TI-99 und Schneider CPCs. 384 Seiten, A4, DM 59,-

# COMPUTER FÜR KINDER



Ein Buch für Kinder und ihre Lehrer – ein kindgemäßes Buch für die erste Be gegnung mit Computern, ihren Eigenwilligkeiten, und ihren unerschöpflichen Möglichkeiten Ein Buch zu unserer Gegenwart und zur Zukunft unserer Kinder "Computer für Kinder" nichtet sich an Kinder im Alter von 8 bis 13 Jahren, für deren Interesse an Computern keines der unzähligen Computer-Bucher ge schneben wurde

"Computer für Kinder" ist ganz auf Kinder eingestellt und be schäftigt sich unterhaltsam und leicht verständlich mit folgenden Themen

Wie arbeiten Computer Wie funktioniert mein Computer

Wie programmiert man mit einfachen Flußdiagrammen

Wie kann ich BASIC leicht verstehen Programme aufbauen mit Befehlen Farbige Graphiken entwerfen Erklärung von Computer-Begriffen

Sally Greenwood Larson war Kindergärtnerin, ehe sie selbst Computern begegnete und zwischen den Weiten von Kindern und Computern zu vermitteln begann.

Computer für Kinder, A4 quer, Fadenheftung, über 100 Seiten, je Ausgabe DM 29,80 vorliegend für: VC 20, C 64, Apple II, Atari

Fordern Sie Unterlagen über Unterlagen ant programm an



DM 29.80



DM 29,80



COMPUTER FÜR KINDER

DM 59,-



DM 199.-



DM 239.-



DM 49



DM 36.-

592 —	<xf></xf>	11040 ENDPROC	<0
593 REM - Player Missiles mach -	<0C>	11100	<3
594 REM - links bewegen -	<nz></nz>	11102 REM - Bild speichern und -	<h< td=""></h<>
599 —	<yh></yh>	11103 REM - Parameter anzeigen -	≺U
600 PROC PMLEFT	<yk></yk>	11104	< K
620 IF PMNR=0	<v5></v5>	11110 PROC BILDSPEICHERN	KH
622 X0POS=X0POS-1 624 IF X0POS<44 THEN X0POS=44	<up></up>	11120 MOVE DPEEK(BB).ADR(SCREEN\$).7680	<6
624 IF X0PUS<44 THEN X0PUS=44	<fr></fr>	11130 ENDPROC	< L
	<0Z>	15000 PROC PARAMANZ	<₽
628 ENDIF	<un></un>	15010 CLS	<0
630 IF PMR=1	<wj></wj>	15020 POSITION 9,3:? "PARAMETER_ANZEIGEN	
632 X1P0S=X1P0S-1	<vm></vm>	B COLUMN TO THE	<b>≺</b> T
	<fi></fi>	15030 POSITION 9,4:7 "	
636 PDKE 53249,X1PQS	<67>		<0
638 ENDIF	<up></up>	15040 POSITION 3.7:? "Linker_Rand_=_":XM	-
790 ENDPROC	<pz></pz>	IN	<0
792 —	CX3>	15050 POSITION 3,9:? "Rechter_Rand_=_";X	
793 REM - Player Missiles mach -	<06>	MAX	< F
			10
794 REM - rechts bewegen	<bm></bm>	15060 POSITION 3,11:? "Unterer_Rand_=_";	10
799 —	<yl></yl>	YMAX	<0
800 PROC PMRIGHT	<zv></zv>	15070 POSITION 3,13:7 "Oberer_Rand_~_";Y	
820 IF PMR=0	<vw></vw>	MIN	<f< td=""></f<>
	<n.></n.>	15090 POSITION 3,15:? "Tiefex=x"; TMAX	<2
	<xe></xe>	15100 GET EIN	<l< td=""></l<>
926 POKE 53248, X0POS	<603>	15110 ENDPROC	<£
828 ENDIF	<800>	15199	<
830 IF PMNR=1	<hin></hin>	15200 REM - Fehlerbehandlung -	<2
832 X1F0S=X1P0S+1	<ui></ui>	15210	<0
834 IF XIPOS>198 THEN X1POS=198	< TM>	20000 REM FEHLERROUTINE	<₽
836 POKE 53249,X1POS	<0M>	20010 ? "(ESC CTL 2)"	<1
838 ENDIF	<ut></ut>	20015 POSITION 3,10:2 "FEHLER_NR"; ERR;	
998 ENDPROC	<0D>	"_IST_AUFGETRETEN"	<0
0002	<36>	20017 POSITION 3,12:7 "ZEILENANR.A"; ERL	< 2
0002 REM - Variablen und Player -	<fn></fn>	20020 IF ERR=170 THEN POSITION 3,20:? "D	
0003 REM - Missiles definieren	KKZ>	ATEI_NICHT_VORHANDEN"	< 5
0004	<jw></jw>	20030 IF ERR-162 THEN POSITION 3,20:? "D	
2009 PROC DECLARE	<vm></vm>	ISKETTE_IST_VOLL"	< €
0010 DIM FILE\$ (20) ,DAT\$ (20) ,SCREEN\$ (768		20040 IF ERR-164 THEN FOSITION 3,20:7 "D	
),LEER\$(256),LD\$(10),RU\$(10),E\$(20)	<wz></wz>	ATEI_NICHT_IN_DRONUND"	< 1
0020 CDL0=40:COL1=202:COL2=148:COL3=0	(AJ)	20050 IF ERR-169 THEN POSITION 3,20:? "D	
0030 LEER\$(1)-"(CTL ,)":LEER\$(256) "(CT		IRECTORY_IST_VOLL"	< 0
·	<im></im>	20060 IF ERR=160 THEN POSITION 3,20:? "D	
,}":LEER*(2)=LEER* 0040 LO\$="{CTL ,}{CTL O}{CTL H}{CTL H}{		ISKETTENLAUFWERK_NICHT_BEKANNT"	< 5
		20070 IF ERR=128 THEN POSITION 3,20:2 "U	
TE HE (CTE HE (CTE HE (CTE ,)":RU#="(CTE ,	1	EBERTRAGUNG_UNTERBROCHEN"	<1
COTE AS COTE AS COTE AS COTE AS COTE			1 )
0) (CTL ,) "	≺EZ>	20080 IF ERR=167 THEN POSITION 3,20:? "D	17
0100 ENDPROC	<ua></ua>	ATELLIST_SCHREIBGESCHUETZT"	<7
1000	<je></je>	20090 IF ERR=144 THEN POSITION 3,20:7 *D	,
1002 REM - Apfelmaennchen auf -	<gf></gf>	ISKETTE_IST_SCHREIBGESCHUETZT"	∢1
1003 REM - dem Bildschirm dar	<c1></c1>	20100 IF ERR=138 THEN POSITION 3,20:2 "G	
1004 REM - stellen -	<uc></uc>	ERAET_NICHT_ANSPRECHBAR"	<1
1005	<kd></kd>	20110 IF ERR-165 THEN POSITION 3,20:? "F	
1010 PROC BILDHOLEN	<ix></ix>	ALSCHER_DATE INAME"	<6
1015 GRAPHICS 15+16	<yr></yr>	20115 POSITION 3,22:? "WEITER_MIT_RETURN	
1020 MOVE ADR (SCREENS) , DPEEK (88) ,7688	<if></if>	4.17	< 8
1030 POKE 708, COLO: POKE 709, COL1: POKE 7		20120 GET EIN: TRAP 40000	<6
0,COL2:POKE 712,COL3	<ib></ib>	20130 GOTO 95	< W







# Grafikspielereien in Turbo-BasicXL

### Mit einfachen Mitteln lassen sich unter Turbo-Basic XL schöne Grafikeffekte erzeugen.

le neuen Befehle von Turbo-Basic XL eröffnen dem Programmierer eine Reihe zusätzlicher Möglichkeiten, die unter dem normalen Atari-Basic nur mit Hilfe von Assembler-Unterroutinen realisierbar waren. So erlaubt auch der Turbo-Basic-Befehl »MOVE«, der Speicherbereiche blitzschnell verschieben kann, für Basic völlig neue Programmtechniken. Anhand eines Grafikbildes zeigt das Programm »Turbo-Basic-Grafikdemo«, wie sich die Daten des Bildschirmspeichers schnell und einfach verschieben oder speichern lassen.

Das Programm ist in einzelne Prozeduren unterteilt, die am Anfang aufgerufen werden. Mit Ausnahme der Prozeduren »DECLARE« und »DEMO« spielt die Reihenfolge des Aufrufs keine Rolle. Die Routinen, die an der Grafik Veränderungen vornehmen, sind in sich abgeschlossen und stellen vor ihrem Ende stets den ursprünglichen Zustand des Bildes her. Die Prozeduren bewirken im einzelnen folgendes:

#### Prozedur »DECLARE«

Initialisieren und dimensionieren der benötigten Stringvariablen.

#### Prozedur »DEMO«

Erzeugen einer Spirale im Grafik 15-Format. An diesem Bild werden alle weiteren Manipulationen vorgenommen. Hier kann man natürlich sowohl andere Grafiken erzeugen als auch ein Bild von Diskette laden.

#### Prozedur »TEXTEIN«

Einblenden eines Textes in das Grafikbild. Der dabei überschriebene Teil der Grafik wird zuvor in einem String gespeichert. So ist sichergestellt, daß man den ursprünglichen Zustand des Bildes wiederherstellen kann.

#### Prozedur »XMIRROR«

Spiegeln der Grafik um die X-Achse. Das Bild wird also auf den Kopf gestellt.

#### Prozedur »XKLAPPT«

Spiegeln der oberen Hälfte des Bildschirms nach unten.

#### Prozedur »XKLAPPB«

Hierbei handelt es sich im Prinzip um die gleiche Routine wie XKLAPPT. Es wird lediglich die untere Hälfte nach oben gespiegelt.

#### Prozedur »XCOPYT«

Kopieren der oberen Bildschirmhälfte nach unten.

#### Prozedur »XCOPYB«

Untere Hälfte des Bildes nach oben kopieren.

#### Prozedur »YCOPYL«

Die linke Hälfte des Bildschirms nach rechts kopieren.

#### Prozedur »YCOPYR«

Die rechte Bildschirmhälfte nach links kopieren.

#### Prozedur »PUTPIC«

Ablegen des Bildschirminhalts in einem String.

#### Prozedur »GETPIC«

Der Inhalt eines Strings wird auf den Bildschirm geschrieben.

#### Prozedur »EFFEKTE«

Die Inhalte der Farbregister 708, 709 und 710 werden ausgetauscht. Dadurch entsteht ein schöner Farbeffekt.

Alle Werte in den beschriebenen Prozeduren beziehen sich übrigens auf ein Bild in Grafikstufe 15. Selbstverständlich können die Effekte auch in allen anderen Grafikstufen verwendet werden. Man muß dann lediglich die Größe des Bildschirmspeichers sowie die Anzahl der benötigten Byte pro Zeile entsprechend anpassen. In Grafikstufe 15 errechnet sich die Bildschirmgröße beispielsweise aus 192 Zeilen und 40 Byte pro Zeile. 192 multipliziert mit 40 ergibt also 7680.

(Wolfgang Czerny/wb)

PROGRAMM-STECKBRIEF				
Programmname	Grafik-Spielerelen			
Programmtyp	Grafik			
Programmlersprache	Turbo-Basic XL			
Programmlänge	7939 Byte			
für Computer	Atan 800 XL/130 XE			
zusätzliche Hardware	Diskettenlaufwerk			
Eingabehilfe	Prüfsummer			
Bemerkung	Effektvolle Grafikmanlpulation			
Leserservice	Diskette SPIELETUR			



Bild 1. Ein Grafik 8-Bild mit Textfenster

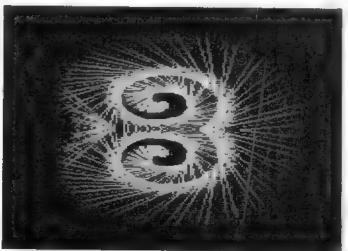


Bild 2. Hier Ist das Textfenster wieder gelöscht und die Grafik von Bild 1 gespiegelt



100 REM SPIELEREIEN MIT DEM	<wt> [</wt>	7099	≺XV
110 REM MOVE BEFEHL IN TURBO BASIC	<2D>	B000 PROC XCOPYT	KPB
115 REM	<qd></qd>	8820 EXEC PUTPIC	KZA
128 REM VON WOLFGANG CZERNY	<xn></xn>	8030 FOR I=0 TO 3848 STEP 40	<nh< td=""></nh<>
130	<31>		KKF
160 EXEC DECLARE	(LK)	8050 NEXT I	<fl.< td=""></fl.<>
176 EXEC DENG	<wv></wv>	8868 PAUSE 500	< MM
			<0K
175 00	<fc></fc>	8070 EXEC GETPIC	
177 EXEC TEXTEIN	<10>	8080 ENDPROC	<6H
180 EXEC XKLAPPT	<kb></kb>	8890 —	< WH
190 EXEC XKLAPPB	<zf></zf>	9000 PROC XCOPYS	KEA
200 EXEC XMIRROR	<rg></rg>	9028 EXEC PUTPIC	< ZB
210 EXEC XCOPYT	<x8></x8>	9030 FOR I=3840 TO 7680 STEP 40	<as< td=""></as<>
220 EXEC XCOPYB	<mt></mt>	9040 MOVE ADR(A\$)+1,DPEEK(88)-3840+1,40	< RQ
230 EXEC YCOPYL	<b>&lt;5Z&gt;</b>	9050 NEXT I	KEM
248 EXEC YCOPYR	<wn></wn>	9060 PAUSE 300	<wx< td=""></wx<>
250 LOOP	<ch></ch>	9070 EXEC BETPIC	< OL
500 END	<x0></x0>	9080 ENDPROC	KPI
777 —	<ld></ld>	7999 —	KYP
1000 PROC DECLARE	<dd></dd>	10000 PROC YCOPYL	KAH
1010 DIM A\$ (8000) ,B\$ (8000) ,LEER\$ (800)	(NC>	10020 EXEC PUTPIC	<nv< td=""></nv<>
1020 A\$(1)="(CTL ,)":A\$(8000)="(CTL ,)":	(1407)	10030 FDR I=0 TD 191	<iz< td=""></iz<>
1020 H*(1)="(C(C +)":H*(8000)="(C C +)":	<hv></hv>		
	75145	10040 MOVE DPEEK(88)+I+40,DPEEK(88)+I+40	<ua< td=""></ua<>
1025 B\$(i)="(CTL ,)":B\$(8000)="(CTL ,)":	ZLVS	+20,20	
9\$ (2) +9\$	<lx></lx>	10050 NEXT I	<gd< td=""></gd<>
(030 LEER\$(1)="(CTL ,)":LEER\$(800)="(CTL		10040 PAUSE 300	<by< td=""></by<>
, > ": LEER\$ (2) = LEER\$	<jv></jv>	18870 EXEC GETPIC	<cl< td=""></cl<>
1040 ENDPROC	<00>	10080 ENDPROC	< VD
1090	<we></we>	10079 —	<mf< td=""></mf<>
	<bk></bk>	10100 PROC YCOPYR	<ei< td=""></ei<>
2010 FXEC PUTPIC	<yr></yr>	10120 EXEC PUTPIC	< NY
2020 MOVE DPEEK(88)+3840,ADR(84),800	<ls></ls>	10130 FOR I=0 TG 191	<30
2030 MOVE ADR(LEER\$), DPEEK(88)+3840,800	<lf></lf>	10140 MOVE DPEEK(88)+I*40+20,DPEEK(88)+I	
2040 TEXT 0,98,"DIES_IST_FIN"	<in></in>	*40,20	KLX
2050 TEXT 0,107,"TEXTFENSTER"	<3U>	10150 NEXT I	<66
2060 PAUSE 400	<we></we>	10160 PAUSE 300	<cb< td=""></cb<>
2070 MOVE ADR (B#) ,DPEEK (88) +3840,800	(SF>	19170 EXEC GETPIC	<c0< td=""></c0<>
2080 PAUSE 400	<mk></mk>	10180 ENDPROC	<v6< td=""></v6<>
2070 EXEC GETPIC	<dk></dk>	10199 —	<mi< td=""></mi<>
2100 ENDPROC	<0F>	11000 PROC GETPIC	<ba< td=""></ba<>
2110 -	<vk></vk>	11010 MOVE ADR (A\$) , DFEEK (88) ,7680	(SR
5000 PROC XMIRROR	<md></md>	11020 ENDPROC	KUH
5010 FOR J=1 TO 10	<di></di>	11999	<ni< td=""></ni<>
5020 EXEC PUTPIC	<yx></yx>	12000 PROC PUTPIC	<s6< td=""></s6<>
	<px></px>		
		12010 MOVE DPEEK(88),ADR(A\$),7680	<ej< td=""></ej<>
5040 MOVE ADR(A\$)+I-40,DPEEK(88)+7690-1,		12020 ENDPROC	< U3
40	<f6></f6>	13000 PROC EFFEKTE	<zz< td=""></zz<>
5050 NEXT I	<fi></fi>	13010 PDKE 764,255	<rh< td=""></rh<>
5060 NEXT J	<fw></fw>	13020 REPEAT	<hm< td=""></hm<>
5070 PAUSE 500	<mm></mm>	13030 POKE 708,40:POKE 709,202	<z3< td=""></z3<>
6075 EXEC BETPIC	<pb></pb>	1.5040 POKE 710,148:PAUSE 5	<ve< td=""></ve<>
5080 ENDPROC	<pe></pe>	13050 POKE 708,202:POKE 709,148	< U.
5090	<m3></m3>	13860 POKE 710,40:PAUSE 5	KNS
5000 PROC XKLAPPT	<e3></e3>	13070 POKE 708,149:POKE 709,40	< KL
020 EXEC PUTPIC	<yy></yy>	13080 POKE 710,202: PAUSE 5	KEP
030 FOR 1=3840 TO 0 STEP -40	<my></my>	13090 UNTIL PEEK(764)<>255	< DJ
040 MOVE ADR (A\$)+1-48, DPEEK (88)+7680-1.		13100 ENDPROC	KUE
19	<lh></lh>	13119	<jf< td=""></jf<>
050 NEXT I	<fj></fj>	15000 PROC DEMO	<p4< td=""></p4<>
070 EXEC EFFEKTE	<ft></ft>	15010 XI=80:YI=70:Q=0:DE6	<if< td=""></if<>
5080 EXEC GETPIC	<0L>	15020 GRAPHICS 31:C=20	₹C5
5090 ENDPROC	<pi></pi>	15040 FOR I=1 TO 1000 STEP 5	<01
5099 —	<xu></xu>	15050 Q=Q+1:IF Q>3.5 THEN Q=1	<3L
	<sq></sq>	15060 CDLOR Q:R=1/10:T=1	<fl< td=""></fl<>
7000 PROC XKLAPPS			
7020 EXEC PUTPIC	<yz></yz>	15070 X=R*COS(T):Y=R*SIN(T)	<00
7030 FOR I=7690 TO 3840 STEP -40	<00>	15080 IF Y+Y1>151 THEN 15140	< BE
7040 MOVE ADR(A*)+1-40,DPEEK(88)+7680-1,		15090 PLOT X+XI,Y+YI	< 1416
40	<li></li>	15100 X=(1+C)/16*COS(I+C+90)	KAE
	KEK>	15110 Y=(I+C)/16*SIN(I+C+90)	< DL
	⟨FU⟩	15120 DRAWTO X+XI,Y+YI	
7050 NEXT I 7070 EXEC EFFEKTE 7080 EXEC GETPIC		15120 DRAWTG X+XI,Y+YI 15130 NEXT I	715> AV>

Listing zu »Turbo-Basic-Grafikdemo«

# Mehr Tempo für Player Missiles

In normalem Basic ist die Positionierung von Playern und Missiles eine zeitaufwendige Angelegenheit, in Turbo-Basic XL dagegen nicht.

er sich schon einmal mit der Player-Missile-Grafik der Atari-Computer auseinandergesetzt hat, der weiß, daß die horizontale Verschiebung recht einfach ist. Die vertikale Positionierung hingegen gestaltet sich schon wesentlich aufwendiger. Das Programm »PMMOVE« demonstriert, wie man den MOVE-Befehl in Verbindung mit der Player-Missile-Grafik sinnvoll einsetzen kann. Die Auswahl, welchen Player man bewegen möchte, erfolgt mit den Tasten 1 bis 4. Und hier eine kurze Programmbeschreibung: Prozedur »DECLARE«

Hier werden die Stringvariablen dimensioniert und vier Strings mit den Werten für die verschiedenen Player versehen. Die Höhe jedes Players beträgt in diesem Fall acht Zeilen. Ober- und unterhalb ist jeweils eine Leerzeile hinzugefügt, damit beim Verschieben der alte Player nicht jedesmal extra gelöscht werden muß.

#### Prozedur »INIT«

Hier werden die Basiswerte für die Player-Missile-Erzeugung festgelegt. Zudem wird der Speicherbereich der Player gelöscht, Auch diese Maßnahme wird mit Hilfe des MOVE- Befehls durchgeführt. Die Anfangspositionen der vier Player sind durch die Felder X() und Y() festgelegt.

#### Prozedur »PM-BAUEN«

Die Strings, die für das Aussehen der Player verantwortlich sind, werden durch den MOVE-Befehl in die entsprechenden Speicherbereiche verlegt. Es folgt dann die Fixierung der horizontalen Positionen.

(Wolfgang Czerny/wb)

PROGRAMM-STECKBRIEF				
Programmname	Player/Missile-Mover			
Programmtyp	Utility			
Programmiersprache	Turbo-Basic XL			
Programmlänge	2372 Byte			
für Computer	800 XL/130 XE			
zusätzliche Hardware	keine			
Eingabehilfe	AMPEL			
Bemerkung	Das Programm zeigt, wie man mit den speziellen Turbo-Basic-Befehlen Player Missiles behandeln kann			
Leserservice	Diskette (PMMOVE.TUR)			

100 REM PLAYER/MISSLE STEUERUNG	<rv></rv>		FOR I=1 TO 10	<ra:< td=""></ra:<>
100 REM PLAYER/MISSLE STEUERUNG 110 REM _ IN TURBO-BASIC 120 REM 130 REM von Wolfgang Czerny 140 REM	<kb></kb>	10120	READ W:LOOK2*(I,I)=CHR*(W)	<rx:< td=""></rx:<>
120 REM	< MA >	10130	NEXT I	<fy< td=""></fy<>
130 REM von Wolfgang Czerny	<ma></ma>	10140	FDR I=1 TO 10	<rm< td=""></rm<>
148 REM	<aq></aq>	10150	READ WILOOKS\$(I,I)=CHR\$(W)	<tb< td=""></tb<>
145	<b><jz></jz></b>	10160	NEXT I	KGK
150 EXEC DECLARE	<li></li>	10170	FOR T=1 TO 10	<ry< td=""></ry<>
155 GRAPHICS 31	<xc></xc>	10180	READ W:LOOK4\$(I,I)=CHR\$(W)	<uf< td=""></uf<>
160 EXEC INIT	<00>>	10170	NEXT I	KGW
148 REM 145 150 EXEC DECLARE 155 GRAPHICS 31 160 EXEC INIT 170 EXEC PM_BAUEN	<cd></cd>	10500	REM LOOK1	<it< td=""></it<>
1000 DO	<tm></tm>	10510	DATA-0,1,2,4,8,144,224,224,240,0	< WC
1010 REPEAT	KED>		REM LOOK2	<30
1020 GET EIN	<1F>		DATA_0,240,224,224,144,8,4,2,1.0	<86
1030 UNTIL (EIN>48 AND EIN<53) OR EIN=42			REM LOOKS	KKN
OR EIN=43 OR EIN=45 OR EIN=61	<di></di>		DATA_0,128,64,32,16,9,7,7,15,0	<zl.< td=""></zl.<>
1040 IF EIN>48 AND EIN<53 THEN NRWEIN-48	<qa></qa>		REM LOOK4	KLK
1050 IF EIN=42 THEN X(NR)=X(NR)+1			DATA-0,15,7,7,9,16,32,64,128,0	<bv< td=""></bv<>
1060 IF EIN=43 THEN X(NR)=X(NR)-1	<qr></qr>		ENDPROC	<wi< td=""></wi<>
1070 IF EIN=45 THEN Y(NR)=Y(NR)-1		10999		<nb< td=""></nb<>
1080 IF EIN=61 THEN Y(NR)=Y(NR)+1	<qf></qf>	de	PROC INIT	<xc< td=""></xc<>
1898 IF Y(NR)>255 THEN Y(NR)=8	<mp></mp>	11010	PMB=PEEK (106) -40: POKE 54279. PMB	<rl< td=""></rl<>
1180 IF Y(NR) (0 THEN Y(NR)=255	<00>	11020	PMB-PMB+256: NR-1	<18U
1110 IF X(NR)>255 THEN X(NR)=0	<km></km>		POKE 559,62: REM SINGLE RES.	<yc< td=""></yc<>
1128 IF X(NR)<8 THEN X(NR)=255	(BP)		FOR I=53256 TO 53259	<pr< td=""></pr<>
1138 POKE 53247+NR.X (NR)	<v0></v0>		POKE I.0	<vk< td=""></vk<>
1140 IF NR=1 THEN MOVE ADR(LOOK1#),PMB+1		_	NEXT I	< GJ
224+Y(1).10	CHXD		FOR 1=704 TO 707	<wv< td=""></wv<>
1150 IF NR=2 THEN MOVE ADR(LOOK2\$),PMB+1	******		POKE I.255	<id< td=""></id<>
2804Y(2) _10	(RT)	_	NEXT I	KGV
1160 IF NR=3 THEN MOVE ADR(LOOK3\$),PMB+1			FOR I=1024 TO 1792 STEP 256	<b>&lt;</b> GJ
536+Y(3) . 10	<by></by>	11119	MOVE ADR (LEER\$) , PMR+1, 256	<ua< td=""></ua<>
1170 IF NR=4 THEN MOVE ADR(LOGK4\$),PMB+1			NEXT I	<fw< td=""></fw<>
	<lu></lu>		X(1)=50:Y(1)=50:X(2)=60	<00
2000 LCOP	<xv></xv>	11150	Y(2)=60: X(3)=70: Y(3)=70	<va< td=""></va<>
2010 END	<pt></pt>		X(4)=8Ø:Y(4)=8Ø	⟨JR
2079	<x@></x@>		ENDPROC	<vm< td=""></vm<>
	<tt></tt>	11170		<mk< td=""></mk<>
10000 PROC DECLARE			PROC PM BAUEN	<ts< td=""></ts<>
	<am></am>		MOVE ADR(LOOK1\$),PM8+1024+Y(1),10	
10030 DIM LOOK1*(10),LOOK2*(10)	(FG)			KEF
and the same transfer of the s	<ik></ik>		MOVE ADR (LOOK2\$), PM8+1280+Y(2),10	
	<cb></cb>		MOVE ADR(LOOK3\$),PMB+1535+Y(3),10	<ka< td=""></ka<>
10060 POKE 730,1	<uf></uf>		MOVE ADR (LOGK4\$) ,PMB+1792+Y(4) ,10	<rw< td=""></rw<>
10070 RESTORE	<00>		FOR I=1 TO 4	<kr< td=""></kr<>
19980 FOR I=1 TO 19	<rz></rz>		POKE 53247+1,X(I)	<zr< td=""></zr<>
	<se></se>		NEXT I	<gp< td=""></gp<>
18100 NEXT I	<fm></fm>	12000	POKE 53277,3	<qy< td=""></qy<>
			ENDPROC	<vl.< td=""></vl.<>

# Daten komprimiert gespeichert

Wer Speicherplatz auf Diskette sparen möchte, der kann in Turbo-Basic XL zum Speichern und Laden von numerischen Daten auch die Befehle » PUT« und » GET« verwenden.

as Atari-Basic kennt zum Speichem und Laden von Daten mehrere Anweisungen. Zum Schreiben benutzt man die Befehle PRINT und PUT, zum Lesen die Befehle INPUT und GET. Unter Turbo-Basic XL allerdings verfügt man neben den Anweisungen BPUT und BGET, die für die Blockspeicherung zuständig sind, auch über die Befehle %PUT und %GET.

In diesem Fall sollen jedoch nur diejenigen Befehle betrachtet werden, mit denen man Gleitkommazahlen auf Diskette speichert oder von Diskette lädt. Wie das Listing zeigt, kann man dazu die Befehle PRINT und INPUT beziehungsweise %PUT und %GET verwenden. Diese beiden Befehlspaare führen zwar dieselbe Operation aus, gehen dabei aber unterschiedliche Wege. Speichert man eine Gleitkommazahl mit Hilfe des PRINT-Befehls, so benötigt jede Zahl 8 Byte Speicherplatz auf Diskette. Verwendet man hingegen für denselben Vorgang den %PUT-Befehl, benötigt eine Gleitkommazahl nur mehr 6 Byte Speicherplatz.

Im Listing werden nun 1 000 Zufallszahlen in einem Feld abgelegt und auf die beiden verschiedenen Methoden auf Diskette gespeichert. Anschließend laden wir sie wieder. Wie man der Tabelle entnehmen kann, spart man bei der Speicherung von nur 1 000 Zahlen mit den %PUT- und %GET-Befehlen immerhin 56 Sektoren auf Diskette. Das sind rund 6,8 KByte. Es versteht sich, daß gespeicherte Zahlen jeweils nur von dem äquivalenten Lese-Befehl gelesen werden können.

Neben dem geringeren Speicherbedarf hat die Verwendung von %PUT und %GET auch noch einen weiteren Vorteil. Die benotigten Schreib- und Lesezeiten verringern sich um etwa 50 Prozent. Die genauen Zeiten sind in der Tabelle festgehalten. Hierbei wurden übrigens drei verschiedene Disketten-Laufwerke verwendet. Ein Atari-810-Laufwerk, das neuere 1050-Laufwerk und ein aufgerüstetes 1050-Laufwerk (wie im Artikel »Rasende Daten« in diesem Heft beschrieben).

(Wolfgang Czerny/wb)

PROGRAMM-STECKBRIEF				
Programmname %PUT und %GET-Demo				
Programmtyp	Demo			
Programmiersprache	Turbo Basic XL			
Programmlänge 1008 Byte				
für Computer alle				
zusätzliche Hardware Diskettenlaufwerk				
Eingabehilfe	Prüfsummer			
Bemerkung				
Leserservice	Diskette (PUTGETTUR)			

Sasic- Befehl	Ausführungszeiten in Sekunden						Benötigte Sektoren
	Erweitertes 1050-Laufwerk		Normales 1050-Laufwerk		Atari- 810-Laufwerk		
	Mit Venty	Ohne Verify	Mit Verify	Ohne Venify	Mit Verify	Ohne Venify	
PRINT	32.74	27.48	58.22	35.32	56.22	34.02	104
INPUT	21.2	20.68	34.48	34.34	32.48	32.48	
%PUT	19.34	16.68	28.98	17.64	27.30	16.50	40
%GET	11.42	11.86	16.28	16.12	15.26	15.26	48

Tabelle mit den Ausführungszeiten unter Verwendung der verschiedenen Befehle, mit denen sich Daten speichern und laden lassen.

10	<yw></yw>
20 REM - %PUT und %SET Demo - 30 REM - von Frank Ostrowski -	(AR)
30 REM - von Frank Ostrowski -	<yf></yf>
48	<yz></yz>
100 7 "XPUT_Demonstration"	<zz></zz>
110 DIM A(1000)	<wx></wx>
120 ? "Creating_Array"	<db></db>
130 FOR I 1 TO 1000	(GZ)
140 A(I)=RND	<fk></fk>
150 NEXT I	<6B>
160 —	<jo></jo>
170 7 "PRINT_File"	<ak></ak>
189	<b>(JS)</b>
190 T=TIME	<p6></p6>
200 OPEN #1,8,0,"D:RNDTST.DAT"	<sx></sx>
210 FOR I =1 TO 1000	<gw></gw>
220 7 #1;A(I)	<uu>&gt;</uu>
230 NEXT I	<fy></fy>
240 CLOSE #1	<lg></lg>
250 ? "_PRINT:_"; (TIME-T) /50; "_Sekunden"	
260	<jp></jp>
270 T-TIME	<dd></dd>
280 OPEN #1,4,0,"D:RNDTST.DAT"	<rn></rn>
290 FOR I=1 TO 1000	<hm></hm>
300 INPUT #1,A:A(I)-A	<rf></rf>
310 NEXT I	<fv></fv>
320 CLOSE #1	<ld></ld>
330 2 "_INPUT: _"; (TIME-T) /50; "_Sekunden"	<ff></ff>
340 DIR "D:RNDTST.DAT"	<mv></mv>
350 DELETE "D: RNDTSY. DAT"	<@X>
360	<jq></jq>
370 ? "_%PUT_File"	<bu></bu>
380 -	<uu></uu>
390 T=TIME	<pi></pi>
400 OPEN #1,8,0,"D:RNDTST.DAT"	<b><sz></sz></b>
410 FOR I=1 TO 1006	<gy></gy>
420 %PUT #1:A(I)	<yn></yn>
438 NEXT I	<6A>
440 CLDSE 01	(LI)
450 ? "_XPUT_:_"; (TIME-T)/50; "_Sekunden"	
460 —	<jr></jr>
478 T=TIME	(PF)
488 OPEN #1,4,6,"D:RNDTST.DAT"	(RP)
490 FOR I=1 TO 1000	<ho></ho>
500 XGET #1,A:A(I)=A	<eg></eg>
510 NEXT I	<fx></fx>
520 CLDSE #1	<lf></lf>
530 2 "_XGET_:_"; (TIME-T) /50; "_Sekunden"	
540 DIR "D:RNDTST.DAT"	
	<mx></mx>
SSØ DELETE "D:RNDTST.DAT"	<qz></qz>
Listing zu = % PUT und % GET« (Turbo-Basic XL)	
Libung 20 - APOT ONO AGET (TOTO-BASIC AL)	

# Ordnung muß sein

»Wer suchet, der findet« ist das Motto vieler Programmsammlungen. Mit »Happy-Disksorter« wird die Verwaltung einer Softwaresammlung zum Kinderspiel. Sehen Sie selbst.

ine Programmsammlung kann so schnell wachsen, daß man den Überblick über seine wertvolle Software verliert. Dann legt man Karteikarten an, oder hält auf losen Zetteln fest, welche Programme sich auf welcher Diskette befinden. Nun, ideal scheinen diese Lösungen nicht. Da sich ein Computer aber auch hervorragend zur Datenspeicherung eignet, liegt es nahe, ihm diese Aufgabe der Programmverwaltung zu übertragen. Dazu benötigt man natürlich spezielle Software.

»Happy-Disksorter« besteht aus zwei Teilen. Erstens dem Maschinenspracheteil (Listing 1) mit dem Menübild und der geänderten Display-List, zweitens dem Hauptprogramm (Listing 2). Gearbeitet wird nur mit dem Hauptprogramm. Listing 1 wird stets hinzugeladen und liegt dann sozusagen im Hintergrund vor.

Bevor Sie mit »Happy-Disksorter« arbeiten können, müssen beide Listings eingetippt sein. Dann laden und starten Sie zuerst Listing 1. Dieses Programm erzeugt das Maschinenfile mit dem Namen »MENU.PIC« auf Diskette. Anschließend brauchen Sie nur noch Listing 2 aufzurufen und zu starten. Es folgt dann automatisch das File »MENU.PIC« und Sie gelangen in das Hauptmenü vom »Happy-Disksorter« (Bild). Beachten Sie, daß sich dieses File unbedingt auf der gleichen Diskette wie das Hauptprogramm befindet. Später brauchen Sie dann nur noch das Hauptprogramm aufzurufen, da das Maschinensprache-File nur einmal auf Diskette geschrieben werden muß.

Befinden Sie sich einmal im Hauptmenü, können Sie schon die ersten Programmnamen eingeben. Die Namen dürfen maximal 25 Zeichen lang sein. Anschließend folgt die Frage nach dem Umfang des Programms in Sektoren und des Indexes zum schnellen Auffinden von Programmen.

Um ins Menü des »Happy-Disksorters« zu gelangen, betätigt man bei der Frage nach dem Programmnamen einfach die RETURN-Taste. Daraufhin erscheint am linken oberen Bildschirmrand ein Pfeil, den man mit den Cursortasten zu dem gewünschten Symbol bewegen kann. Die nochmalige Betätigung der RETURN-Taste führt die Funktion aus.

#### Suchen

Man gibt entweder die Eintragsnummer, mit der jedes Programm versehen wird, oder einen unverwechselbaren Teil des gesuchten Programmnamens ein. Sollte ein Eintrag mit dem Suchbegriff übereinstimmen, wird er auf dem Bildschirm ausgegeben. Durch Drücken der RETURN-Taste gelangt man wieder zum Ausgangsmenü zurück.

#### Listen

Die eingetragenen Namen in abfallender Reihenfolge auf dem Bildschirm ausgeben. Ist der Bildschirm mit Namen gefüllt, wird nach Betätigung der Leertaste die nächste Seite auf dem Bildschirm ausgegeben. Mit der ESC-Taste gelangt man jederzeit zurück zum Hauptmenü. Die Ausgabe eines bestimmten Eintrags bewirkt die RETURN-Taste.

#### Sortieren

Nach dem Index oder dem Namen sortieren.

#### Eintrag löschen

Die Auswahl der zu löschenden Daten erfolgt wie unter »Suchen«. Der gesuchte Programmname erscheint dann auf dem Bildschirm und wird durch Drücken der Taste J gelöscht. Disketten-Menü

Mit diesem Symbol gelangt man in ein weiteres Menü:

- Auflisten des Directorys in Laufwerk 1. Der Listvorgang läßt sich mit der START-Taste oder CONTROL-1 unterbrechen. Am Ende gelangt man mit einer beliebigen Taste zurück ins Untermenü.
- Speichem der im RAM-Speicher vorliegenden Programmnamen. Es genügt zur Erkennung des Dateinamens die ersten acht Zeichen einzugeben. Die RETURN-Taste bringt den Benutzer wieder zurück ins Untermenü.
- 3. Laden einer auf Diskette gespeicherten Programmliste. Die Eingabe erfolgt wie unter Punkt 2 beschrieben.
- 4. Alle Eintragungen einer zu katalogisierenden Diskette werden automatisch in die im RAM-Speicher vorliegende Datei übernommen. Durch Drücken irgendeiner Taste, ausgenommen 1 bis 4, gelangt man zum Hauptmenü zurück.

#### Liste drucken

Es besteht die Wahl zwischen einem zwei- und dreispaltigem Ausdruck. Als Drucker ist ein Epson FX80 vordefiniert. Auch hier gelangt man mit Betätigung irgend einer Taste zum Hauptmenü zurück.

#### Fehlermeldungen«

Sollte ein Fehler auftreten, springt das Programm automatisch ins Hauptmenü zurück. Wählt man anschließend die Funktion »Fehlermeldungen«, wird der Fehler auf dem Bildschirm beschrieben.

(Andreas Kubelka/wb)

Zeile 7040 - Die Zeichen, die zwischen Anführungszeichen stehen, bewirken einen Ausdruck im normalen Modus. Diese Einstellung ist für zweispeltigen Ausdruck mit 80 Zeilchen pro Zeile verantnormalen Modus. Diese Einstellung ist für zweispeltigen Ausdruck mit 80 Zeilchen pro Zeile verantnormalen Modus.

Zeile 7100 - Umschalbung auf Ausdruck mit 132 Zeichen pro Zeile und in drei Spalten. Es müssen also stets die Zeichen zwischen Anführungszeichen geändert werden. Am boston orientieren Sie sich an ihrem Druckerhandbuch.

PROGRAMM-STECKBRIEF		
Programmame	Happy-Disksorter	
Programmtyp	Utility	
Programmiersprache	Atari-Basic	
Programmlänge	10034 Byte	
für Computer	800 XL/130 XE	
zusatzliche Hardware	Diskettenlaufwerk	
Eingabehilfe	Prüfsummer	
Bemerkung	Das Programm besteht aus zwei Teilen. Zuerst Listing 1 starten und anschließend Listing 2.	
Leserservice	Diskette (DISKSORT.BAS/ DISKMENU.BAS)	



Das »Happy-Disksorter«-Menū

# TIPS & TRICKS

Zelie	Beschreibung
10 bis 200	Initialisierung
300 bis 390	Eingabe des Namens mit Anzahl der Sektoren und dem Index
400 bis 440	Menű-Auswahl
1000 bis 1050	Auflisten der Namen
1500 bis 1510	Suchen eines Namens aus der Liste
2000 bis 2110	Name in der gesamten Liste suchen
2500 bis 2510	Eingabe des gesuchten Namens
3000	Parameter setzen für Listen in abfallender Reihenfolge
4000 bis 4040	Sortierroutine
5000 bis 5200	Name löschen
6000 bis 6050	Disketten-Untermen@
6100 bis 6140	Schreiben des Directorys
6200	Eingabe des Filenamens
6410 bis 6440	Liste abspeichern/laden
6500 bis 6530	Übernahme des Directorys in die Liste
7000 bis 7130	Druckerausgabe
7400	Name in Kanal Di schreiben
7500 bis 7510	Namennummer in Kanal Di schreiben
7600	Überschrift in Kanal D: schreiben
8000 bis 9200	Fehlermeidung

### Programmbeschreibung

Einsprungadresse	en für Maschinenroutiner
Adresse 38786	String search
Adresse 38925	Bubble Sort

Variable	Beschreibung
US	-enthält die Adresse der Maschinenroutine zum Laden und Speichern
D\$	-String für alte Daten
A\$	—Nimmt Eingaben entgegen
Р	-Position des Pfeils beim Auflisten
C	Zāhler
D	—Enthält Kanalnummer, in den geschrieben wird Kanal . 0 = Bildschirm Kanal . 1 ≃ Drucker
ER	—Fehlercode
TD	-Enthält die zu bearbeitende Position in D\$
L	—Länge von A\$
I .	Für Schleifen und Pfeilposition beim Menü
A	-Nimmt Eingaben über GET entgegen
CNT	—Anzahl der Daten in D\$
STA	Die Startnummer beim Listen oder die aktuelle Position
STO	-Stopposition beim Listen
PLMI	—Gibt die Listrichtung an
F	—Gibt beim Suchen die Position an, an der die Eingabe gefunden wurde
R	Restwert für die Druckerroutine, um die Startnummer der nächsten Spalte zu finden.
S2	—Startnummer der 2. Spalte beim Drucken von 3 Spalten

#### **Variablenliste**

100 GRAPHICS 0: POKE 710,0: OPEN #1,4,0,"K	
:":POKE 752,1	KHE>
110 POSITION 3,10:7 "DISKETTE_IN_LAUFWER	
KA1AUNDARETURN"	<ol></ol>
120 GET #1,A:CLOSE #1:OPEN #1,8,0,"D:MEN	
U.PIC"	<0Z>
130 POSITION 5,12:2 "SCHREIBE_FILE_==>_D	
IMENU. PIC"	<h2></h2>
140 TRAP 380	<ks></ks>
150 FOR I=1 TO 1214	<mb></mb>
160 READ A: B=B+A	<wd></wd>
170 PUT #1,A:IF I/43=INT(I/43) THEN POKE	(NO)
40485+P,PEEK(40485+P)+128:P=P+1	<ai></ai>
180 NEXT I	<gh></gh>
300 POSITION 10,14: IF B<>91431 THEN ? ".	NOM2
ADATA FEHLER"	(DO)
	<rs></rs>
310 IF B=91431 THEN ? "FILE_GESCHRIEBEN" 320 ? :? :END	<hh></hh>
	<mp></mp>
1000 DATA_216, 104, 104, 133, 204, 104, 133, 20	
3,104,133,209,104,133,208,104,133,215,16	
4,133,214,104,104,133,205,104,104	<bm></bm>
1010 DATA_133,206,169,0,133,212,169,0,13	
3,213,162,0,160,0,177,214,224,0,208,2,13	
2,216,209,20B,20B,43,232,228,206	<05>
1020 DATA_240,22,200,196,205,240,50.72.1	
52,72,138,168,177,214,133,207,104,168,10	
4,165,207,24,144,219,72,165,204	<yu></yu>
1030 DATA 133,213,165,203,133,212,104,16	
2,0,224,0,240,17,224,0,240,6,160,0,177,2	
14,162,0,164,216,200,196,205,208	<de>&gt;</de>
1040 DATA_186,165,208,24,101,205,133,208	(2)
,144,2,230,209,165,203,208,6,165,204,240	
,7,198,204,198,203,24,144,156,96	<xx></xx>
1050 DATA-216,104,56,233,3,133,217,104,1	`^^/
33 204 104 133 203 104 127 215 104 122 5	
33,204,104,133,203,104,133,215,104,133,2	(Page 1)
14,104,133,210,104,133,207,162	<rm></rm>
1060 DATA_0,104,104,157,0,1,232,228,217,	
208,246,56,165,209,233,2,133,209,165,210	
,233,0,133,210,48,108,165,209,133	<20>
1070 DATA_211,165,210,133,212,165,204,13	
3,206,133,208,165,203,133,205,24,101,214	
,133,207,165,208,101,215,133,208	<tt></tt>
Heat and the man and the second	
Listing 1 zu »Happy-Disksorter«. Bitte mit dem Prüfs	ummer ein

1080 DATA_160,0,185,0,1,170,2,1,134,218,	
190,1,1,200,200,200,132,216,169,136,177,	
205,209,207,240,12,165,218,208	<tp></tp>
1090 DATA_4,144,16,176,46,144,44,176,10,	
200,202,208,234,164,216,196,217,208,210,	
198,211,169,255,197,211,208,6,166	<dn></dn>
1100 DATA-212,240,11,198,212,165,208,133	420147
,206,165,207,24,144,172,165,213,240,4,13	
4,213,208,148,96,134,213,160,0	<mw></mw>
1110 DATA_177,205,170,177,207,145,205,13	21 H4 L
8,145,207,200,176,214,208,241,240,203,0,	
0,2,0,0,0,0,0,0,0,120,0,0,0,0	<ge></ge>
1120 DATA_0,62,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0	1027
,8,0,0,0,0,0,0,0,0,0,2,0,0,1,1,1,1,1,1,1,1	
0,56,0,0,0,0,54,0,0,0,0,112,0	<uv></uv>
1130 DATA_0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0	1047
0,0,0,2,0,0,0,32,0,0,0,0,56,0,0,0,0,15,6	
2,0,0,0,0,56,0,0,0,0,0,0,0,0,0	<hv></hv>
1140 DATA_0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,64	ZINA
,15,192,0,0,72,15,192,0,0,13,182,0,0,0	
,56,0,0,0,31,255,252,0,0,15,255	/V0\
	<ys></ys>
1150 DATA_240,0,0,0,0,0,0,0,0,0,15,128,0,0	
,128,8,64,0,0,128,8,64,0,0,15,54,240,0,0	Z DUL.
,127,255,248,0,0,24,31,252,0,0	<ph>&gt;</ph>
1160 DATA_13,255,176,0,0,30,0,0,0,0,0,0,0,7,	
0,0,1,0,8,64,0,1,0,8,64,0,0,13,129,192,0	(DV)
,0,127,255,249,0,0,24,31,240,0	<py></py>
1170 DATA 0,15,255,240,0,0,127,128,0,0,0	
,7,226,0,0,18,1,248,127,240,2,1,248,127,	cours
240,0,15,8,192,0,0,12,204,192,0	<qv></qv>
1180 DATA_0,24,31,240,0,0,13,255,176,0,0	
,255,192,0,0,0,4,32,0,0,28,1,8,0,16,4,1,	20V S
8,0,16,0,0,8,192,0,0,12,204,192	<88.
1190 DATA_0,0,24,31,252,0,0,15,255,240,0	
,1,243,224,0,0,0,4,32,0,0,28,1,8,0,16,8,	28823
1,8,0,16,0,0,9,240,0,0,12,204,192	<nk></nk>
1200 DATA_0,0,31,255,252,0,0,13,255,176,	
0,1,225,224,0,0,0,4,63,248,0,30,63,15,25	41.5
4,16,16,63,15,254,16,0,0,8,0,0	<lf></lf>
1218 DATA_0,12,204,172,0,0,31,247,252,0,	
0,15,255,240,0,0,3,224,0,0,0,4,0,9,0,0,3	41150
3,0,2,16,0,33,0,2,16,0,0,62,0,0	<hc></hc>
1220 DATA 0,12,204,192,0,8,31,227,252,0, ngeben.	
iigeveii.	

	0,13,255,176,0,0,7,192,0,0,0,4,0,8,0,0,3	
	3,0,2,16,0,33,0,2,16,0,0,28,0,0 1238 DATA_0,12,204,192,0,0,31,193,252,0,	<rz></rz>
1	0,15,255,240,0,0,15,128,0,0,0,4,0,8,0,0,	
I	33,255,194,16,0,33,255,194,16,0 1240 DATA_0,8,0,0,0,12,204,192,0,0,31,22	<tx></tx>
I	7,252,0,0,15,255,240,0,0,31,0,0,0,0,4,0,	
I	8,0,0,32,0,67,240,0,32,0,67,240 1250 DATA.0,0,0,0,0,0,12,204,192,0,0,31,	<6n>
I	247,252,0,0,0,0,0,0,6,62,0,6,0,0,4,0,8,0	
I	,0,32,0,66,0,0,32,0,66,0,0,0,0 1260 DATA.0,0,0,12,204,192,0,0,31,255,25	<eb></eb>
I	2,0,0,127,255,254,0,0,62,7,223,124,0,4,0	
I	,8,0,0,32,0,66,0,0,32,0,66,0,0 1270 DATA_0,0,0,0,0,12,204,192,0,0,31,24	<is></is>
ı	7,252,0,0,96,0,6,0,0,0,6,216,96,0,7,255,	
ŀ	248,0,0,32,0,126,0,0,32,0,126,0 1280 DATA.0,15,188,240,0,0,12,204,192,0,	<05>
١	0,31,227,252,0,0,127,255,254,0,0,62,7,22	
I	3,124,0,0,0,0,0,0,32,0,64,0,0,32 1290 DATA.0,64,0,0,13,182,192,0,0,12,204	<l6></l6>
Į	,192,0,0,31,227,252,0,0,127,255,254,0,0,	
	62,6,195,12,0,0,0,0,0,0,32,0,64 1300 DATA_0,0,32,0,64,0,0,15,188,192,0,0	<ca></ca>
	,12,204,192,0,0,31,227,252,0,8,127,255,2	
	54,0,0,62,5,223,124,0,0,0,0,0,0,0 1310 DATA_63,255,192,0,0,63,255,192,0,0	<1T>
	13,182,192,0,0,12,204,192,0,0,31,247,232	
ŀ	,0,0,127,255,254,0,0,0,0,0,0,0,0 1320 DATA_0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0	<mm></mm>
I	13,188,240,0,0,15,255,192,0,0,31,255,252	
ŀ	,0,0,12,0,48,0,0,0,0,0,0,52,79 1330 DATA_195,152,15,15,15,15,15,15,15,1	<bd></bd>
ŀ	5,15,15,15,15,15,15,15,15,15,15,15,15,66	
	,64,156,2,2,2,2,2,2,2,2,112 1340 DATA.Z,112,2,2,2,2,2,2,2,2,2,2,65,1	<kn></kn>
١	1,156	<10>
I	Listing 1 zu »Happy-Disksorter« (Schluß)	

10 OPEN #2,4,0,"K:"	<bt></bt>
28 US=ADR(=) (CTL ,) (CTL M)/(CTL B) Xh"(CT	12.11
L P) Z(CTL C)1(CTL C)(CTL M)R(CTL C)N(CT	
L MALKETE CANKETE MATKETE CANKETE MAYKET	
L CONCETT MOXCETT COLVETCETT POOCETT LOC	
ETL MORKETE COLVED . (CTL MO/KETL B) (CTL .	
)"):P=7	<nz></nz>
40 OPEN #1,4,0,"D:MENU.PIC": I=USR(US,387	
86,1214)	<yu></yu>
50 POKE 708,10:POKE 709,10:POKE 710,0:PD	
KE 82.0: POKE 560,11	<dc></dc>
200 DIM D\$(24000),A\$(25)	<iw></iw>
300 TRAP 300:POKE 559,34:CLOSE #1:POP :E	
R=PEEK(195):POKE 195,0:? "(ESC CTL ()(ES	
C CTL => (ESC CTL => ":LD=LEN(D\$)	<ad></ad>
310 POKE 752,0:POKE 39977+P,2:POKE 39979	
+P,2:? "Name_:_";:INPUT #16;A\$:L=LE	
N(A\$)	<bl></bl>
320 IF L=0 THEN 400	<un></un>
325 IF L<25 THEN FOR I=L+1 TO 25:A\$(I,I)	
="_":NEXT I	<6C>
327 D\$(LD+1,LD+25) A\$:D\$(LD+26,LD+30)="0	
O'O"	<bs></bs>
330 POSITION 1,5:? "Sektor##::";:INPUT	
#16; A\$: L=LEN(A\$): IF L=0 OR L>3 THEN ? " (	
ESC CTL 2)";:GOTO 330	<ib></ib>
350 D\$(LD+29-LEN(A\$),LD+28)-A\$	<dx></dx>
36@ PDSITION 4,7:7 "Index.:.";:INPUT #16	
;A\$1 IF LEN(A\$) <2 THEN ? "KESC CTL 2)"; :G	
010 368	<jn></jn>
380 D\$(LD+29,LD+30)=A\$(1,2)	(CJ)
390 GOTO 300	<mu></mu>
400 POKE 752,1:2 "(ESC CTL <)":I-2	<k6></k6>
Listing 2 zu »Happy-Disksorter«.	
Bitte mit dem Prüfsummer eingeben.	
(Fortsetzung auf Seite 74.)	
(Fortactzung auf Seite 74.)	



H P Blomeyer-Bartenstein/R Both Datenkommunikation und Lokale Computer-Netzwerke

2. uberarbeitete Auflage Februar 1985, 284 Sexten Das Thema Datenkommunikation und Rechnerate (Tolematik) wird mit zu-nehmender Computensierung unserer Gesellschaft immer mehr an Bedeutung gewinnen. Wer die Entwicklung der Zukunft nicht verpassen will, sollte sich daher in dieses wichtige Thema rechtzeitig einarbeiten. Dazu bietet das vor-liegende Buch die beste Möglichkeit. ingenne Bisch die beste Moglichkert.
Von Spezialisten geschrieben und dennoch lercht verstandlich fuhrt es in das komplexe Gebet ein. Für entsprechend vorgebildete Leser werden auch die physikalischen und mathematischen Zusammenhänge erlaitert.
Best.-Nr. MT 790 120M 3.89100-1.79.9.

DM 58,-ISBN 3-89090-079-8



W. Jakobsen

Lexikon der modernen Elektronik Juni 1985, 280 Seiten

Die zweite überarbeitete und erweiterte Auflage dieses Standard-Werks bringt austührliche Erfaurerungen zu ca. 3000 englischsprachigen Fachbegriffen aus engischsprachger ractivegram aus der Weit der allgemeinen Elektronik, Militroelektronik, Computertechnik und Software. Das deutsch-englische Stichwortverzeichnis und die Querverweise zu den einzelnen Sochwörtern machen es zum idealen Nachschlagewerk für Beruf, Ausbildung und Hobby: Best-Nr MT 752 ISBN 3-89090-0804

DM 52,-



P Rosenbeck

Personal Computer Lexikon 2. uberarbeitete Aultage Oktober 1985, 200 Seiten

Dieses Lexicon mit seinen mehr als 1700 Stichworten, mit gut lesbaren Erklärungen und Hintergrundinformstionen, mit einem ausgearbeiteten Ver-weisungssystem und der Gegenüber-stellung englischer und deutscher Fachbegriffe stellt ein unentbehrliches Nachschlagewerk und Hilfsmitterfür die Arbeit mit Microcomputern dar

Best-Ny. MT 758

DM 32.-



M-P Gottloh/G Streeker

BTX professionell eingesetzt

\$1 x proressionen eingesetzt
1984, 287 Seiten
Wer sich mit dem Gedanken trägt, selbst
8TX-Anbieter zu werden, dieses neue
Medium in ingendener Weise beruflich
und geschäftlich zu nutzen, der benötigt
weitergehende Informationen, als sie eine Errführung zu geben vermag. Das dezu nötige Wissen vermittet das vertiene Errüfthrung zu geben vermitag. Das dan zu nötige Wisson vermittelt das vörlie-gende Buch. Es hilft Ihnen, völlig neue Möglichkeiten in Marketing und Wer-bung, bei Dienstleistungen, bei der Infor-mationsdistribution und bei Schulungen zu erschließen, lourzi das neue Medium BTX professionell das heißt kostensen-nand und derschleistend absentation. kend und gewinnbringend einzusetzen

BTX professionell nutzen! Best-Nr. MT 530

DM 68.-ISBN 3-922120-62-0

Markt & Technik-Fachhücher erhalten Sie bei fierem Bochhändler.



m-Presel-Shade 2, 2013 Blaze hai bii



*** PORTY ON T 4.0 BEECO COOLEGE OT -1		I -NEG BARRY TARNA -N RA- AFEC FECTAFEC	
410 POSITION I,1:? "(ESC ESC) (ESC CTL -)		L =>"1? "(ESC TAB)4=>=Dir_(ESC ESC)(ESC	ZWES
":GET #2,A:POSITION I,1:? "_": I=I+5*(A=4			<me></me>
2 AND I(37)-5*(A-43 AND I)2)	<ra></ra>	6040 IF A>48 THEN ON A-48 GOTO 6100,6200	4 7 14 2
430 IF A=155 AND (LD>0 OR I=27 OR I=37)		,6200,6500	<jx></jx>
THEN CNT=LD/30:ON (I-2)/5+1 GOTG 2000,10		6050 GOTO 300	<01>
00,3000,4000,5000,6000,7000,8000	<uc></uc>	6100 OPEN #1,4,0,"D: #. #"	<kr></kr>
440 GOTO 410	<ng></ng>	6118 ? "(ESC CTL <)":? :? ,"FilenameEx	
1000 STA=CNT:STO-0:PLMI=-1	<ip></ip>	tacasaLaenge":? ,"(CTL M)(CTL M)	
1010 D-0:C=0:? *{ESC CTL {}*:GOSUB 7600:	7-17	CETE MY CETE MY CETE MY CETE MY CETE MY LECT	
		L M) (CTL M) (CTL M) (CTL M) (CTL M) (C	
? "(ESC CTL =)(ESC CTL =)":POKE 39977+P.	414040		400 LIS
112: POKE 39979+P,112	<ym></ym>		<5V>
1020 GOSUB 7400: 7	<mk></mk>	6120 INPUT #1; As: IF As(2,2) <>""" THEN ?	
1030 STA-STA+PLMI	<sx></sx>	:? ,A\$:GET #2,A:CLOSE #1:GOTO 6000	<pt></pt>
1035 IF STA-STO THEN GOSUR 1100:60TO 300	<rv></rv>	6130 IF PEEK(53279)=6 THEN 6130	<bv></bv>
1040 IF PEEK(84)=19 THEN GOSUB 1100:GOTO		6140 POSITION 0,4:? "(ESC SHIFT DEL)":PO	
1010	<hn></hn>	SITION 10,17:? A\$(3,10),A\$(11,13),A\$(15,	
			<ac></ac>
1050 6010 1020	<od></od>	17):60T0 6120	VHC2
1100 POSITION 3,P+4:? "(ESC ESC) (ESC CTL		6200 POKE 752,0:? "(ESC CTL =) (ESC CTL =	
*>":GET #Z,A:IF A=32 THEN RETURN	(NS>	>Filename_?_{CTL Y} (8_Zeichen) (CTL Y)ia"	
1103 IF A=43 OR A=42 THEN STA=STA-PLMI:S		::INPUT #16;A\$:POKE 752,1:L=LEN(A\$):IF L	
TO=0+(CNT+1) + (STO=0): PLMI=-PLMI: RETURN	<mn></mn>	=0 OR L>8 THEN 6000	<00>
1105 IF A=27 THEN 300	<ax></ax>	6418 As(11)=As(1,L):As(1,2)="D:":As(3)=A	
1107 IF A=155 THEN 1500	<nc></nc>		<ba:< td=""></ba:<>
	<nw></nw>	\$(11,10+L):A\$(3+L)=".STR"	
1108 POSITION 3,P+4:7 "A"	(1441)	6420 IF A=50 THEN OPEN #1,8,0,A#:? #1;LD	(KU)
1110 P=P+1+(A=61 AND P(14)-1+(A=45 AND P		6430 IF A=51 THEN OPEN #1,4,0,A#: INPUT #	
>0):POSITION 3,P+4:? "(ESC ESC) (ESC CTL		1; LD: D\$="_": D\$ (LD-1) =D\$: D\$ (2) =D\$	<xa></xa>
#3"	<eq></eq>	6440 I=USR(US,ADR(D\$),LD):GOTO 6000	<b><jb></jb></b>
1120 FOR I=1 TO 5:POKE 39975+I+P,2+110*(		6500 POKE 559,0: OPEN #1,6,0,"D: *. *"	(LB)
I=2 OR I=4):NEXT 1:GOTO 1100	<rn></rn>	6518 INPUT #1; A\$: IF A\$(2,2) <>"" THEN PO	
1500 LD=0:POSITION 3,P+4:FOR I=0 TO 2:?		KE 559,34:CLOSE #1:GOTO 6000	<di></di>
"(ESC CTL +)";:LD=LD+10^1+(PEEK(93)-16):		6520 D\$(LD+26,LD+28)=A\$(15,17):A\$=A\$(3,1	72/2/
NEXT I:LD=(LD-1)+30:? "(ESC CTL ()"	<00>		
	<is></is>	8):A\$(9)="":D\$(LD+1,LD+	1004
1505 IF LDC THEN 300	1137	25) =A\$: D\$(LD+29) ="**"	(SY)
1510 POSITION 12,3:7 D*(LD+1,LD+25); "(ES		6530 LD=LD+30:GOTO 6510	<mm></mm>
C TAR) (FSC TAB) (ESC TAB) (ESC CTL =) (ESC		7000 ? "(ESC CTL =)(ESC CTL =)(ESC CTL =	
CTL *> (ESC CTL *> (ESC CTL *> (ESC CTL *> (		)(ESC CTL =)(ESC CTL =)(ESC TAB)11==>_Dr	
ESC CTL *}";D\$(LD+26,LD+28);"(ESC CTL =)		ucke_2_Spalten(ESC CTL =)":? "(ESC TAB)2	
(ESC CTL =) (ESC CTL +) (ESC CTL +) (ESC CT		_==>_Drucke_3_Spalten"	<ph></ph>
L +3";D\$(LD+29,LD+38):PGSITION 0,3:GOTO		7020 GET #2,A:IF A<49 OR A>50 THEN 300	<vn></vn>
310	<mr></mr>	· ·	/AI4>
		7030 D=1:C=0:OPEN #D,4,0,"P:":IF A=50 TH	
2000 TRAP 2020: GOSUB 2500: GOTO 2030	<00>	EN 7100	(SJ)
2020 GOSUB 2100	<xp></xp>	7040 7 #D; "(ESC ESC)! (CTL ,)": GOSUB 7600	
2030 STA=STA-P: IF STA(1 THEN STA=1	<yg></yg>	:GOSUB 7600:? #D:? #D	< X V >
2040 STO=CNT+1:PLMI=1:60T0 1010	(LF)	7045 STO=INT((CNT-0.1)/2)+1	<ne></ne>
2100 F=USR(38786,CNT,ADR(D\$),ADR(A\$),30,		7050 FOR I=1 TO CNT	KTK>
LEN(A\$)): IF F-0 THEN 300	(BV)	7060 STA=I:GOSUB 7400:STA=STO+I:GOSUB 74	
2110 STA-CNT-F+12RETURN	<mu></mu>		<b><gt></gt></b>
2500 ? * (ESC CTL <) (ESC CTL =) (ESC CTL =	(110)	7100 7 #D: "(ESC ESC)!(CTL D)":GOSUB 7600	1017
> (ESC TAB)Gebe_Term_oder_Nummer_ein (ESC		:GOSUB 7600:GOSUB 7600:7 #D:7 #D	<aq></aq>
CTL =>":? "(ESC TAB)===>,";:INPUT #16;A\$			<tn></tn>
:STA=VAL(A\$):IF STA(1 OR STA)CNT THEN 38		7120 FOR I=1 TO CNT:S2=I+STO+1*(R>0)	くばしと
0	<uk></uk>	7:30 STA=I:GOSUB 7400:STA=S2:GOSUB 7400:	
2510 RETURN	<ph></ph>	STA=S2+STO+1*(R)0.5):GOSUB 7400:? #D:NEX	
3000 STA=1:STO=CNT+1:PLMI=1:GOTO 1010	<3D>	TI	<wu></wu>
4000 ? "(ESC CTL ()"	<kt></kt>	7400 GOSUB 7500:? #D;A\$;"4";D\$(LD-29,LD-	
4010 POSITION 0,6:? "(ESC TAB)1=>_Namen		5); "AA"; D\$ (LD-4, LD-2); "A"; D\$ (LD-1, LD); "A	
_sortieren(ESC CTL =)":2 "(ESC TAB)2.=>_			/EAS
COG INDIZATIA		A";:RETURN	<fa></fa>
Index_enripres(ESC CTL =\N.CCT #2 A.DOV		TERR C-CALLE EXCUST THEN C. MR. C. COC. C.	
<pre>Index_sortieren(ESC CTL =)*:GET #2,A:POK E 550 A</pre>	(B7)	7500 C=C+1: IF C>CNT THEN ? #D:CLOSE #1:6	4.7.7.5
E 559,0	<8Z>	OTO 300	<tz></tz>
E 559,0 4020 IF A=49 THEN I=USR(38925,ADR(D*),30		0TO 300 7510 A*="000":A*(4-LEN(STR*(STA)))=STR*(	
E 559,0 4020 IF A=49 THEN I=USR(38925,ADR(D*),30 ,CNT,1,28,0,29,2,0)	<bz></bz>	0TO 300 7510 A*="000":A*(4-LEN(STR*(STA)))=STR*(	<tz></tz>
E 559,0 4020 IF A=49 THEN I=USR(38925,ADR(D*),30 ,CNT,1,28,0,29,2,0) 4030 IF A=50 THEN I=USR(38925,ADR(D*),30		0TO 300 7510 A*="000":A*(4-LEN(STR*(STA)))=STR*(	
E 559,0 4020 IF A=49 THEN I=USR(38925,ADR(D*),30 ,CNT,1,28,0,29,2,0)		0TO 300 7510 A*="000":A*(4-LEN(STR*(STA)))=STR*( STA):LD-STA*30:RETURN 7600 ? #D; "Nr-aaaa>>>>Programm<	
E 559,0 4020 IF A=49 THEN I=USR(38925,ADR(D*),30 ,CNT,1,28,0,29,2,0) 4030 IF A=50 THEN I=USR(38925,ADR(D*),30	<ze></ze>	OTO 300 7510 A*="000":A*(4-LEN(STR*(STA)))=STR*( STA):LD-STA*30:RETURN 7600 ? #D; "Nr-aaaa>>>>Programma<<< <aa>Caaaa&gt;&gt;&gt;&gt;Programma</aa>	<x1></x1>
E 559,0 4020 IF A=49 THEN I=USR(38925,ADR(D*),30 ,CNT,1,28,0,29,2,0) 4030 IF A=50 THEN I=USR(38925,ADR(D*),30 ,CNT,29,2,0,1,28,0)	<ze></ze>	OTO 300 7510 A*="000":A*(4-LEN(STR*(STA)))=STR*( STA):LD-STA*30:RETURN 7600 ? *D; "Nr>>>Programm<<<< Sek_Ina_";:RETURN 8000 TRAP 9200:POKE 39982,7:POSITION 0,9	<xi></xi>
E 559,0  4020 IF A=49 THEN I=USR(38925,ADR(D*),30 ,CNT,1,28,0,29,2,0)  4030 IF A=50 THEN I=USR(38925,ADR(D*),30 ,CNT,29,2,0,1,28,0)  4040 POKE 559,34:60T0 300  5000 TRAP 5020:60SUB 2500:60T0 5030	<ze></ze>	OTO 300 7510 A*="000":A*(4-LEN(STR*(STA)))=STR*( STA):LD-STA*30:RETURN 7600 ? *D; "Nr>>> Programm< < Sek_Ina_";:RETURN 8000 TRAP 9200:POKE 39982,7:POSITION 0,9 :? "a_";:GOSUB 9000+ER	<x1></x1>
E 559,0  4020 IF A=49 THEN I=USR(38925,ADR(D*),30 ,CNT,1,28,0,29,2,0)  4030 IF A=50 THEN I=USR(38925,ADR(D*),30 ,CNT,29,2,0,1,28,0)  4040 POKE 559,34:60T0 300  5000 TRAP 5020:60SUB 2500:60T0 5030  5020 GOSUB 2100	<ze></ze>	0TO 300 7510 A*="000":A*(4-LEN(STR*(STA)))=STR*( STA):LD-STA*30:RETURN 7600 ? *D; "Nr-AAAA>>>>Programma<<<<	<xi></xi>
E 559,0  4020 IF A=49 THEN I=USR(38925,ADR(D*),30 ,CNT,1,28,0,29,2,0)  4030 IF A=50 THEN I=USR(38925,ADR(D*),30 ,CNT,29,2,0,1,28,0)  4040 POKE 559,34:60T0 300  5000 TRAP 5020:60SUB 2500:60T0 5030  5020 GOSUB 2100  5030 ? "(ESC CTL =)":60SUB 7400:? :? "(E	<ze></ze>	OTO 300 7510 A*="000":A*(4-LEN(STR*(STA)))=STR*( STA):LD-STA*30:RETURN 7600 ? *D; "Nr-AAAA>>> Programma<<<	<xi><av><xj></xj></av></xi>
E 559,0  4020 IF A=49 THEN I=USR(38925,ADR(D*),30 ,CNT,1,28,0,29,2,0)  4030 IF A=50 THEN I=USR(38925,ADR(D*),30 ,CNT,29,2,0,1,28,0)  4040 POKE 559,34:60T0 300  5000 TRAP 5020:60SUB 2500:60T0 5030  5020 GUSUB 2100  5030 ? "(ESC CTL *)":GOSUB 7400:? :? "(ESC CTL *)":GOSUB 7400:? !? "(ESC CTL *)":GOSUB 7400:? "(ESC CTL *)"	<ze></ze>	OTO 300 7510 A*="000":A*(4-LEN(STR*(STA)))=STR*( STA):LD-STA*30:RETURN 7600 ? *D; "Nr-AAAA>>>>Programm<  ASEKAINAA";:RETURN 8000 TRAP 9200:POKE 39982,7:POSITION 0,9 1? "AAA";:GOSUB 9000+ER 8002 ? :? "(ESC TAB)(ESC TAB)(ESC TAB)(E SC TAB)DrueckeaeineaTaste":SET #2,A:POKE 39982,2:POKE 195,0:GOTO 300	<xi> <av> <xj> <km></km></xj></av></xi>
E 559,0  4020 IF A=49 THEN I=USR(38925,ADR(D*),30 ,CNT,1,28,0,29,2,0)  4030 IF A=50 THEN I=USR(38925,ADR(D*),30 ,CNT,29,2,0,1,28,0)  4040 POKE 559,34:60TO 300  5000 TRAP 5020:60SUB 2500:60TO 5030  5020 GUSUB 2100  5030 ? "(ESC CTL *)":GOSUB 7400:? 1? "(E SC CTL *)(ESC CTL *) (ESC CTL *)(ESC CTL *)	<ze> <va> <sq> <tq> <xs></xs></tq></sq></va></ze>	OTO 300 7510 A*="000":A*(4-LEN(STR*(STA)))=STR*( STA):LD-STA*30:RETURN 7600 ? *D; "Nr>>> Programm <<<	<xi> <av> <xj> <km> <sa></sa></km></xj></av></xi>
E 559,0  4020 IF A=49 THEN I=USR(38925,ADR(D*),30 ,CNT,1,28,0,29,2,0)  4030 IF A=50 THEN I=USR(38925,ADR(D*),30 ,CNT,29,2,0,1,28,0)  4040 POKE 359,34:60TO 300  5000 TRAP 5020:60SUB 2500:60TO 5030  5020 GOSUB 2100  5030 ? "(ESC CTL *)":GOSUB 7400:? :? "(E SC CTL *) (ESC CTL *) (ESC TAB) (ESC CTL *) (ESC CTL *) (ESC CTL *) Name Loeschen (J/N ) ":GET *2,4:IF A<>74 THEN 300	<ze> <va> <sq> <tq> <xs> <tq> <xs></xs></tq></xs></tq></sq></va></ze>	OTO 300 7510 A*="000":A*(4-LEN(STR*(STA)))=STR*( STA):LD-STA*30:RETURN 7600 ? *D; "Nr>>>>Programm<(<	<xi> <av> <xj> <km></km></xj></av></xi>
E 559,0  4020 IF A=49 THEN I=USR(38925,ADR(D*),30 ,CNT,1,28,0,29,2,0)  4030 IF A=50 THEN I=USR(38925,ADR(D*),30 ,CNT,29,2,0,1,28,0)  4040 POKE 559,34:60TO 300  5000 TRAP 5020:60SUB 2500:60TO 5030  5020 GOSUB 2100  5030 ? "(ESC CTL =)":60SUB 7400:? :? "(E SC CTL =) (ESC TAB) (ESC CTL *) (ESC CTL *) (ESC CTL *) Name=Loeschen=(J/N ) ":GET *2,A:IF A<>74 THEN 300  5100 IF STA=CNT THEN 5200	<ze> <va> <sq> <tq> <xs></xs></tq></sq></va></ze>	OTO 300 7510 A*="000":A*(4-LEN(STR*(STA)))=STR*( STA):LD-STA*30:RETURN 7600 ? *D; "Nr>>> Programm <<<	<xi> <av> <xj> <km> <sa></sa></km></xj></av></xi>
E 559,0  4020 IF A=49 THEN I=USR(38925,ADR(D*),30 ,CNT,1,28,0,27,2,0)  4030 IF A=50 THEN I=USR(38925,ADR(D*),30 ,CNT,29,2,0,1,28,0)  4040 POKE 359,34:60T0 300  5000 TRAP 5020:60SUB 2500:60T0 5030  5020 GOSUB 2100  5030 ? "(ESC CTL ±)":60SUB 7400:? :? "(E SC CTL =)(ESC CTL ±) ":60SUB 7400:? :? "(E SC CTL ±)(ESC CTL ±) "Amme_Loeschen_(J/N)":6ET #2,A:IF A<>74 THEN 300  5100 IF STA=CNT THEN 5200  5110 D\$(STA*30-29,CNT*30 30)=D\$(STA*30+1)	<ze> <va> <sq> <tq> <xs> <tq> <xs></xs></tq></xs></tq></sq></va></ze>	OTO 300 7510 A\$="000":A\$(4-LEN(STR\$(STA)))=STR\$( STA):LD-STA*30:RETURN 7600 ? #D; "Nr-aaaa>>>>Programma<<< <a>aSekaInaa";:RETURN 8000 TRAP 9200:POKE 39982,7:POSITION 0,9 :? "aaa";:GOSUB 9000+ER 8002 ? :? "(ESC TAB)(ESC TAB)(ESC TAB)(E SC TAB)DrueckearineaTaste":SET #2,A:POKE 39982,2:POKE 195,0:GOTO 300 9000 ? "aaKEINaFEHLER":RETURN 9137 ? "aaFILEaFEHLER":RETURN 9138 ? "aGERAETaFEHLT":RETURN</a>	<xi> <av> <xj> <km> <sa> <px></px></sa></km></xj></av></xi>
E 559,0  4020 IF A=49 THEN I=USR(38925,ADR(D*),30 ,CNT,1,28,0,29,2,0)  4030 IF A=50 THEN I=USR(38925,ADR(D*),30 ,CNT,29,2,0,1,28,0)  4040 POKE 559,34:60TO 300  5000 TRAP 5020:60SUB 2500:60TO 5030  5020 GOSUB 2100  5030 ? "(ESC CTL ±)":60SUB 7400:? :? "(E SC CTL =) (ESC TAB) (ESC CTL #) (ESC CTL #) (ESC CTL #) )":GET #2,A:IF A<>74 THEN 300  5100 IF STA=CNT THEN 5200	<ze> <va> <sq> <tq> <xs> <tq> <xs></xs></tq></xs></tq></sq></va></ze>	OTO 300 7510 A*="000":A*(4-LEN(STR*(STA)))=STR*( STA):LD-STA*30:RETURN 7600 ? *D; "Nr-AAAA>>>>Programma<< ASEKAIDAA";:RETURN 8000 TRAP 9200:POKE 39982,7:POSITION 0,9 1? "AAA";:GOSUB 9000+ER 8002 ? :? "(ESC TAB)(ESC TAB)(ESC TAB)(ESC TAB)(ESC TAB)DrueckearineaTaste":SET *2,A:POKE 39982,2:POKE 195,0:GOTO 300 9000 ? "AAKEINAFEHLER":RETURN 9137 ? "AFILEAFEHLER":RETURN 9138 ? "AGERAETAFEHLT":RETURN 9144 ? "FEHLERAIMAGERAET":RETURN	<xi> <av> <xj> <xm> <xm> <xm <xm="" <xm<="" td=""></xm></xm></xm></xj></av></xi>
E 559,0  4020 IF A=49 THEN I=USR(38925,ADR(D*),30 ,CNT,1,28,0,29,2,0)  4030 IF A=50 THEN I=USR(38925,ADR(D*),30 ,CNT,29,2,0,1,28,0)  4040 POKE 559,34:60T0 300  5000 TRAP 5020:60SUB 2500:60T0 5030  5020 GOSUB 2100  5030 ? "(ESC CTL ±)":60SUB 7400:? :? "(E SC CTL =>(ESC CTL ±) ":60SUB 7400:? :? "(E SC CTL ±) ":60SUB 7400:? "(E SC CTL ±) ":60SUB 7400:? "(E SC CTL ±) ":6	<ze> <va> <sq> <tq> <xs> <tq> <xs></xs></tq></xs></tq></sq></va></ze>	OTO 300 7510 A*="000":A*(4-LEN(STR*(STA)))=STR*( STA):LD-STA*30:RETURN 7600 ? *D; "Nr-AAAA>>>>Programma<<<<	<xi> <av> <xj> <xi> <xj> </xj></xi></xj></av></xi>
E 559,0  4020 IF A=49 THEN I=USR(38925,ADR(D*),30 ,CNT,1,28,0,27,2,0)  4030 IF A=50 THEN I=USR(38925,ADR(D*),30 ,CNT,29,2,0,1,28,0)  4040 POKE 559,34:60T0 300  5000 TRAP 5020:60SUB 2500:60T0 5030  5020 GOSUB 2100  5030 ? "(ESC CTL =)":GOSUB 7400:? :? "(ESC CTL =) (ESC TAB) (ESC CTL *) (ESC CTL =) (ESC CTL =) Name Loeschen (J/N) )":GET #2,A:IF A<>74 THEN 300  5100 IF STA=CNT THEN 5200  5110 D*(STA*30-29,CNT*30 30)=D*(STA*30+1,CNT*30)	<ze> <va> <sq> <tq> <xs> <tq> <xs></xs></tq></xs></tq></sq></va></ze>	OTO 300 7510 A*="000":A*(4-LEN(STR*(STA)))=STR*(STA):LD-STA*30:RETURN 7600 ? *D;"Nr-AAAA>>> Programma<>>> ASEKAINAN;:RETURN 8000 TRAP 9200:POKE 39982,7:POSITION 0,9 :?"AAN;:GOSUB 9000+ER 8002 ? :?"(ESC TAB)(ESC TAB)(ESC TAB)(ESC TAB)DrueckeaeineaTaste":SET *2,A:POKE 39982,2:POKE 195,0:GOTO 300 9000 ? "AKEINAFEHLER":RETURN 9137 ? "AGERAETAFEHLER":RETURN 9138 ? "AGERAETAFEHLER":RETURN 9144 ? "FEHLERAIMAGERAET":RETURN 9163 ? "ADISKETTEAVOLL":RETURN	<xi> <av> <xj> <xj> <km> <sa> <px> <px> <px> <px> <hq> <hq> <hq> <hq> <hq> <hq> <hq> <hq< td=""></hq<></hq></hq></hq></hq></hq></hq></hq></px></px></px></px></sa></km></xj></xj></av></xi>
E 559,0  4020 IF A=49 THEN I=USR(38925,ADR(D*),30 ,CNT,1,28,0,29,2,0)  4030 IF A=50 THEN I=USR(38925,ADR(D*),30 ,CNT,29,2,0,1,28,0)  4040 POKE 559,34:60T0 300  5000 TRAP 5020:60SUB 2500:GOTO 5030  5020 GOSUB 2100  5030 ? "(ESC CTL *)":GOSUB 7400:? :? "(E SC CTL *)(ESC CTL *)(ESC TAB)(ESC CTL *) (ESC CTL *)(ESC CTL *)Name_Loeschen_(J/N) )":GET *2,A:IF A<>74 THEN 300  5100 IF STA=CNT THEN 5200  5110 D*(STA*30-29,CNT*30 30)=D*(STA*30+1,CNT*30) 5200 D*(CNT*30-29)="":CNT*CNT-1:GOTO 500	<ze> </ze>	OTO 300 7510 A*="000":A*(4-LEN(STR*(STA)))=STR*(STA):LD-STA*30:RETURN 7600 ? *D;"Nr-AAAA>>> Programm<	<xi> <av> <xj> <xj> <a href="#"> &lt;</a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></xj></xj></av></xi>
E 559,0  4020 IF A=49 THEN I=USR(38925,ADR(D*),30 ,CNT,1,28,0,29,2,0)  4030 IF A=50 THEN I=USR(38925,ADR(D*),30 ,CNT,29,2,0,1,28,0)  4040 POKE 559,34:60T0 300  5000 TRAP 5020:60SUB 2500:60T0 5030  5020 GOSUB 2100  5030 ? "(ESC CTL *)":GOSUB 7400:? :? "(E SC CTL *)(ESC CTL *)(ESC TAB)(ESC CTL *) (ESC CTL *)(ESC CTL *)Name_Loeschen_(J/N) )":GET *2,A:IF A<>74 THEN 300  5100 IF STA=CNT THEN 5200  5110 D*(STA*30-29,CNT*30 30)=D*(STA*30+1,CNT*30)  5200 D*(CNT*30-29)="":CNT*CNT-1:GOTO 500 0 6000 ? "(ESC CTL <){ESC CTL *)(ESC CTL =)	<ze> </ze>	OTO 300 7510 A\$="000":A\$(4-LEN(STR\$(STA)))=STR\$( STA):LD-STA*30:RETURN 7600 ? *D;"Nr-AAAAA>>>>Programma<<<<	<xi> <av> <xj> <xj> </xj></xj></av></xi>
E 559,0  4020 IF A=49 THEN I=USR(38925,ADR(D*),30 ,CNT,1,28,0,29,2,0)  4030 IF A=50 THEN I=USR(38925,ADR(D*),30 ,CNT,29,2,0,1,28,0)  4040 POKE 559,34:60TO 300  5000 TRAP 5020:60SUB 2500:60TO 5030  5020 GUSUB 2100  5030 ? "(ESC CTL ±)":GOSUB 7400:? 1? "(E SC CTL =)(ESC CTL ±)(ESC TL *) (ESC CTL *)(ESC CTL *)Name_Loeschen_(J/N) )":GET #2,A:IF A<>74 THEN 300  5100 IF STA=CNT THEN 5200  5110 D*(STA*30-29,CNT*30 30)=D*(STA*30+1 ,CNT*30)  5200 D*(CNT*30-29)="":CNT=CNT-1:GOTO 500 0  6000 ? "(ESC CTL <)(ESC CTL =)(ESC CTL = )(ESC CTL =)(ESC CTL = )(ESC CTL =)(ESC CTL = )(ESC CTL = )(	<ze> </ze>	OTO 300 7510 A\$="000":A\$(4-LEN(STR\$(STA)))=STR\$( STA):LD-STA*30:RETURN 7600 ? *D; "Nr-AAAA>>>>Programm<  "SEKAINAA";:RETURN 8000 TRAP 9200:POKE 39982,7:POSITION 0,9 :? "AAA";:GOSUB 9000+ER 8002 ? :? "(ESC TAB)(ESC TAB)(ESC TAB)(E SC TAB)DrueckearineaTaste":SET #2,A:POKE 39982,2:POKE 195,0:GOTO 300 9000 ? "AKEINAFEHLER":RETURN 9137 ? "AFILEAFEHLER":RETURN 9138 ? "AGERAET_FEHLER":RETURN 9144 ? "FEHLERAIMAGERAET":RETURN 9162 ? "ADISKETTEAVOLL":RETURN 9165 ? "FILENAMEAFEHLER":RETURN 9167 ? "FILEAGESCHUEZT":RETURN 9169 ? "DIRECTORYAVOLL":RETURN	<xi> <av> <xj> <xj> </xj></xj></av></xi>
E 559,0  4020 IF A=49 THEN I=USR(38925,ADR(D*),30 ,CNT,1,28,0,27,2,0)  4030 IF A=50 THEN I=USR(38925,ADR(D*),30 ,CNT,29,2,0,1,28,0)  4040 POKE 359,34:60TO 300  5000 TRAP 5020:60SUB 2500:60TO 5030  5020 GOSUB 2100  5030 ? "(ESC CTL *)":GOSUB 7400:? :? "(E SC CTL *)(ESC CTL *)":GOSUB 7400:? :? "(E SC CTL *)(ESC CTL *)NAMELLOSCCTL *)  (ESC CTL *)(ESC CTL *)NAMELLOSCCTL *)  **SET **2,A:IF A<>74 THEN 300  5100 IF STA=CNT THEN 5200  5110 D*(STA*30-29,CNT*30 30)=D*(STA*30+1 ,CNT*30)  5200 D*(CNT*30-29)="":CNT*CNT-1:GOTO 500  60000 ? "(ESC CTL <)(ESC CTL =)(ESC CTL = )(ESC ETL = )(ESC CTL = )(ESC C	<ze> </ze>	7510 A*="000":A*(4-LEN(STR*(STA)))=STR*( STA):LD-STA*30:RETURN 7600 ? *D; "Nr-AAAA>>>>Programma / ASEKAINAM";:RETURN 8000 TRAP 9200:POKE 39982,7:POSITION 0,9 :? "AAA";:GOSUB 9000+ER 8002 ? :? "(ESC TAB)(ESC T	<xi> <av> <xj> <xj> <a href="Text-"> <a hr<="" td=""></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></xj></xj></av></xi>
E S59,0  4020 IF A=49 THEN I=USR(38925,ADR(D*),30 ,CNT,1,28,0,29,2,0)  4030 IF A=50 THEN I=USR(38925,ADR(D*),30 ,CNT,29,2,0,1,28,0)  4040 POKE 359,34:60TO 300  5000 TRAP 5020:60SUB 2500:60TO 5030  5020 GOSUB 2100  5030 ? "(ESC CTL =)":GOSUB 7400:? :? "(E SC CTL =)(ESC CTL =)(ESC TAB)(ESC CTL *) (ESC CTL *)(ESC CTL *)Name_Loeschen_(J/N )":GET *2,A:IF A<>74 THEN 300  5100 IF STA=CNT THEN 5200  5110 D*(STA*30-29,CNT*30 30)=D*(STA*30+1 ,CNT*30)  5200 D*(CNT*30-29)="":CNT=CNT-1:GOTO 500 0  60000 ? "(ESC CTL -)(ESC CTL =)(ESC CTL = }(ESC CTL -)(ESC C	<ze> </ze>	OTO 300 7510 A*="000":A*(4-LEN(STR*(STA)))=STR*(STA):LD-STA*30:RETURN 7600 ? *D; "Nr-AAAA>>>>Programma<<<<	<xi> <av> <xj> <xi> <xj> </xj></xi></xj></av></xi>
E 559,0  4020 IF A=49 THEN I=USR(38925,ADR(D*),30 ,CNT,1,28,0,29,2,0)  4030 IF A=50 THEN I=USR(38925,ADR(D*),30 ,CNT,29,2,0,1,28,0)  4040 POKE 559,34:60TO 300  5000 TRAP 5020:60SUB 2500:60TO 5030  5020 GOSUB 2100  5030 ? "(ESC CTL *)":GOSUB 7400:? :? "(E SC CTL *)(ESC CTL *)":GOSUB 7400:? :? "(E SC CTL *)(ESC CTL *)NAMELLOSCHEN_(J/N) )":GET *2,A:IF A<>74 THEN 300  5100 IF STA=CNT THEN 5200  5110 D*(STA*30-29,CNT*30 30)=D*(STA*30+1 ,CNT*30)  5200 D*(CNT*30-29)="":CNT=CNT-1:GOTO 500  60000 ? "(ESC CTL *)(ESC CTL *)(ESC CTL = )(ESC CTL *)(ESC CTL *)(ESC CTL = )(ESC CTL *)(ESC CTL *)(ESC CTL *)	<ze> </ze>	OTO 300 7510 A*="000":A*(4-LEN(STR*(STA)))=STR*(STA):LD-STA*30:RETURN 7600 ? *D; "Nr-AAAA>>>>Programma /a **Sekalnam";:RETURN 8000 TRAP 9200:POKE 39982,7:POSITION 0,9  *? "AAA";:GOSUB 9000+ER 8002 ? :? "(ESC TAB)(ESC TAB)(ESC TAB)(ESC TAB)DrueckearineaTaste":SET *2,A:POKE 39982,2:POKE 195,0:GOTO 300 9000 ? "AAKEINAFEHLER":RETURN 9138 ? "AGERAETAFEHLER":RETURN 9144 ? "FEHLERAIMAGERAET":RETURN 9144 ? "FEHLERAIMAGERAET":RETURN 9163 ? "ASYSTEMAFEHLER":RETURN 9164 ? "FILEAGESCHUEZT":RETURN 9167 ? "FILEAGESCHUEZT":RETURN 9170 GOTO 9165	<xi> <av> <xj> <xj> <a href="Text-"> <a hr<="" td=""></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></xj></xj></av></xi>
E S59,0  4020 IF A=49 THEN I=USR(38925,ADR(D*),30 ,CNT,1,28,0,29,2,0)  4030 IF A=50 THEN I=USR(38925,ADR(D*),30 ,CNT,29,2,0,1,28,0)  4040 POKE 559,34:60TO 300  5000 TRAP 5020:60SUB 2500:60TO 5030  5020 GOSUB 2100  5030 ? "(ESC CTL =)":GOSUB 7400:? :? "(ESC CTL =)(ESC TAB)(ESC CTL *) (ESC CTL =)(ESC CTL =)\Name_aLoeschen_a(J/N))":GET *2,A:IF A<>74 THEN 300  5100 IF STA=CNT THEN 5200  5110 D\$(STA*30-29,CNT*30 30)=D\$(STA*30+1,CNT*30)  5200 D\$(CNT*30-29)="":CNT=CNT-1:GOTO 500 0  6000 ? "(ESC CTL <)(ESC CTL =)(ESC CTL =)  (ESC CTL =)(ESC CTL =)  (ESC CTL -)(ESC CTL =)  (ESC CTL -)(ESC CTL =)  (ESC CTL -)(ESC CTL -)  (ESC CT	<ze> </ze>	OTO 300 7510 A*="000":A*(4-LEN(STR*(STA)))=STR*(STA):LD-STA*30:RETURN 7600 ? *D; "Nr-AAAA>>>>Programma<<<<	<xi> <av> <xj> <xi> <xj> <a href="#"> <a <="" href="#" td=""></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></xj></xi></xj></av></xi>
E S59,0  4020 IF A=49 THEN I=USR(38925,ADR(D*),30 ,CNT,1,28,0,29,2,0)  4030 IF A=50 THEN I=USR(38925,ADR(D*),30 ,CNT,29,2,0,1,28,0)  4040 POKE 359,34:60TO 300  5000 TRAP 5020:60SUB 2500:60TO 5030  5020 GOSUB 2100  5030 ? "(ESC CTL =)":GOSUB 7400:? :? "(E SC CTL =)(ESC CTL =)(ESC TAB)(ESC CTL *) (ESC CTL *)(ESC CTL *)Name_Loeschen_(J/N )":GET *2,A:IF A<>74 THEN 300  5100 IF STA=CNT THEN 5200  5110 D*(STA*30-29,CNT*30 30)=D*(STA*30+1 ,CNT*30)  5200 D*(CNT*30-29)="":CNT=CNT-1:GOTO 500 0  60000 ? "(ESC CTL -)(ESC CTL =)(ESC CTL = }(ESC CTL -)(ESC C	<ze> </ze>	OTO 300 7510 A*="000":A*(4-LEN(STR*(STA)))=STR*(STA):LD-STA*30:RETURN 7600 ? *D; "Nr-AAAA>>>>Programma<<<<	<xi> <av> <xj> <xi> <xj> <a href="#"> <a <="" href="#" td=""></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></xj></xi></xj></av></xi>

### Submission – ein Weg voller Gefahren

Böse Zungen behaupten, es gibt keine Helden mehr. Falsch! Auf gar mancher Diskette eines Atari 800XL/130XE kämpfen sie für das hohe Ideal der Freiheit.

ur die Besten werden überleben. Unter diesem Motto gilt es sich zu beweisen, denn in den tiefen Gängen des Labyrinths im Spiel »Submission« lauert der Tod mit seinen tausend Gesichtern.

Ziel der Mission ist die Befreiung einer Frau. Der Weg zu ihr führt durch verschiedene Ebenen, die durch Leitern miteinander verbunden sind. Das Klettern ist dabei noch das einfachste, aber es wäre kein Action-Spiel, gäbe es nicht viele Gefahren, die einem das Leben schwermachen. Da plagt man sich mit Krokodilen, Spinnen und Minen, auch an Selbstschußanlagen fehlt es nicht. Unterwegs kann man Goldschätze aufsammeln, für die es natürlich Punkte gibt. Progelungener Befreiungsaktion gibt es dann einen Bonus. Die Abwechslung kommt nicht zu kurz, denn je nach Belieben stehen sieben verschiedene Schwierigkeitsgrade zur Verfügung. Ein weiterer Pluspunkt ist die leichte Handhabung des Spiels. Die einzelnen Spielvarianten ändert man über die SELECT-Taste. Durch Drücken der START-Taste ist man gleich

mitten im Geschehen und nur noch auf die meisterliche Handhabung des Joysticks angewiesen.

»Submission« ist zu 100 Prozent in Maschinensprache geschrieben. Daraus ergibt sich auch die Schnelligkeit des Spielablaufs. Bevor Sie jedoch das Listing zu »Submission« eingeben, müssen Sie unbedingt den AMPEL aus diesem Sonderheft abtippen. Das erleichtert Ihnen die Eingabe von Atari-Maschinensprach-Programmen. Automatisch werden sämtliche Tippfehler erkannt und auf dem Bildschirm angezeigt. (Peter Raab/wb)

Peter Blümer, der Autor des Spiels »Submission«, ist unseren Lesern schon bekannt. In Happy-Computer, Ausgabe 1/85, veröffentlichten wir bereits ein anderes, allerdings in Basic programmiertes, Spiel »Die Schatzhöhle«. Mittlerweile ist für ihn das Wort Maschinensprache kein Fremdwort mehr, wie das Spiel »Submission« beweist. Die Umsetzung von der Idee zum tatsächlichen Spiel ist vorzüglich gelungen. Obwohl das zugehörige Listing entsprechend lang ist, wird sich der Aufwand, es in den Atari einzugeben, wirklich lohnen.

PROG	RAMM-STECKBRIEF
Programmname	Submission
Programmtyp	Spiel
Programmiersprache	Assembler
Programmlänge	11 688 Byle
für Computer	800 XL/130 XE
zusätzliche Hardware	Diskettenlaufwerk oder Kassetten-Recorder
Eingabehilfe	AMPEL
Bemerkung	Labyrinth-Spiel
Leserservice	Diskette (SUBMISS.COM)

	_	_			_	_		_					_									_	_			_
0000:FF	FF	88	96	BA	86	A9	00<4F>	1	0130:02	102	02	02	02	02	22	02<32> [		0260:03	20	96	88	99	88	99	88 <e5></e5>	
8888: 8D	44	82	85	41	8D	C6	02<22>		0138:02	02	02	02	02	82	02	02<3A>		0249:00	99	89	99	90	88	86	00<6C>	
0018:8D	C5	02	9D	C8	02	AD	30(E8)		0140:01	803	03	03	03	23	83	Ø3<41>		0270:00	92	82	92	82	02	92	@2<73>	
0018:02	85	CE	AD	31	02	85	CF<72>	1	D148:03	193	193	83	03	03	933	PI3<40>		0278:02	02	02	82	82	92	82	82<7C>	
0028: A0	ØĐ	A9	46	91	CE	CB	A9(A2)	1	0150:03	03	62	03	23	20	1212	88C2D>		0280:02	92	92	92	81	63	93	03<82>	
0028:57	91	CE	CB	A9	86	91	CE <ca></ca>		0158:00									0288:03	83	83	83	63	03	93	@3<8C>	
6626°C8	A9	86	91	CE	CB	A9	@7<36>	1	2160:00	66	800	5160	202	22	88	00<62>		0290:03	63	83	83	83	03	93	B3<94>	
2038:91-	CE	CB	A9	26	91	CE	CB<8F>		0168:00	20	88	2012	00	60	99	86<4A>		9299:03	98	99	99	99	9F	90	A@<38>	
0048: A9	86	91	CE	A9	36	80	C4<00>	ŀ	0170:00	88	88	99	20	22	6969	00<72>		82AB: 88	9F	1961	96	80	86	98	28<4C>	
2248:22	A9	01	85	99	AØ	50	A2<39>		8178:00	66	20	26	99	98	22	88<7A>		92A8: 90	02	02	82	61	98	98	Ø8<95>	
0050:06	A9	96	4C	50	E4	DB	EE<43>	1	0180:00	20	60	99	90	2021	88	00KB2>		02B0:00	9F	90	AB	20	9F	66	<80.588	
0058:C7	02	4C	5F	E4	22	00	00 <ba></ba>	1	0188:00	66	22	88	22	88	90	00<8A>		9298:90	00	20	80	98	63	<b>Ø</b> 3	@3 <d1></d1>	
0060: 00	89	99	2¢	2F	21	24	29<92>		0170:00	99	99	99	20	92	88	00<92>		02C0±03	22	6969	26	99	88	98	00<46>	
0068: 2E	27	20	99	98	99	90	88<89>		0198:00	22	20	202	88	(36)	99	00<9A>		0209:00	20	22	20	90	F8	23	F3<25>	
8879:88	90	66	20	98	98	88	00<70>		01 A0:00	660	88	80	88	90	22	00 <a2></a2>		8200:24								
0078:00	90	90	98	90	88	88	00<78>		01A8:00	20	20	1343	96	44	44	<b>00&lt;42&gt;</b>		02D8:01								
0080:00	00	99	20	98	82	99	96<86>		0180:47	48	02	02	02	<b>Ø</b> 2	02	02 <e6></e6>		82E9:90								
008B:00	890	F3	F5	E2	ED	E9	F3 <bd></bd>		0199:01	90	20	20	22	2025	86	ØB<25>		0268:00	_							
0090:F3	E9	EF	EE	88	88	99	88<81>		01C0:00	45	45	00	C9	48	93	83<31>		82F0:00								
2078: 82	08	26	88	88	88	80	<b>88&lt;98&gt;</b>		0100:03	83	83	62	62	FC	22	F7 <e0></e0>		02FB: 00								
00A0:00	90	88	98	86	66	90	20 <a2></a2>		@1D0:23	99	88	0.2	88	88	610	Ø8(64)		8266:81								
00AB:00	60	20	20	99	88	96	22 <ca></ca>		01D8:00	45	45	90	4B	4C	62	82<62>		8388:88								
00B0:39	80	30	25	34	25	32	96<2E>		01E0:02	82	102	102	01	60	60	88 <cc></cc>		0310:00		-						
0088:22	20	35	25	20	25	32	00<2E>		01E8:00	90	00	98	80	46	46	20 <bc></bc>		0318:00								
00C0:00	E2	62	E3	92	99	06	9C<9E>		SIF9: CE									0320:00								
88CB: 88	QD	90	86	22	98	22	FB <sc></sc>		0168:03	02	92	02	01	88	20	22<65>		0328:00								
00D8:22	AB	SF	A2	E4	A9	26	26 <cd></cd>		0200:06									0330:0E								
0008:5C	E4	4C	72	49	66	22	68>68		0208:00	22	22	90	99	62	<b>0</b> 3	83<21>		8336:00				-				
99E9:99									0210:03									0340:00								
00EB:00	99	98	98	99	22	99	99 <e8></e8>		0218:00	88	88	20	88	20	22	22<1C>		0348:00								
88F 2: 88									0220:00	62	82	02	01	99	90	@@<@D>		0350:00								
99F8:90	90	96	99	99	98	98	00 <f8></f8>		0228:00	90	60	99	20	88	90	00<2C>		0358:02								
0100:00									0230:00			-	-					0360:02								
0108:00									0238:03									<b>9</b> 299:82								
0110:00	90	100	99	99	20	20	00<12>		0240:00	88	22	98	80	98	90	88<44>		0370:03								
2118:00									0248:00									<b>0378:0</b> 3								
0120:08	80	98	98	88	88	90	99<22>		0250:00									6288:86								
0128:00	80	98	20	66	02	02	02(38)	1	0258:00	98	99	88	98	622	62	83<71>		Ø388: <b>Ø</b> Ø	88	88	98	90	88	88	A8(2E)	
							Lieti	200	zu "Suba	nice	ion	" p	itte	mis	L A B	APEL einge	ab	0.0								
							Listi	ıA	ZU #QUDII	1155	3011	w. D	ILLE	14111	MII	ar LL emge	-	E710.								

0928:23 34 08 34 2F 00 23 28(FB) 0928:21 2E 27 25 00 33 34 21(E3> 29 2E 27 88 33 35<34> 8938:32 34 **9938:22** (NO 20 (845) 20 202 PHO: 28<5B> 094B: 00 88 1200 PIRI 88 88 20 @@K52> 99<7C **0748:00** DO: 6363 1200 11 20 99 0950:00 80 88 22 **00**<62) 0958:00 22 88 88 98 22 013 88<6A> 0960:00 198 90 88 OLC) 210 CADA MA<72> 33< 70> MOAR-MA 75.00 32 75 33 33 99 00<9D> 0970:34 34 34 2F 21 200 32 Ø978:30 20 21 39 88 2E 37<9F> 21 00<5C> 0980:00 27 ZD 25 98 2121 0988:00 00 MM(9A) 88 86 88 D(P) 66 0770:00 00 20 22 20 20 7807 BB(A2) 8998:88 88 88 20 20 25 6969 (BOKAA) 00<B2> 09A0: 00 00 80 88 10/0 1963 (363) 22 20 88 89 BB(BA) 89AB: 88 99 883 8988:88 88 66 88 88 99(C2) 8988: A9 88 BD 25 82 BD P35 M24713 49 29C2:8D C8 82 8D C6 82 46(DD) 29(D3> BD 2F D4 **A2** 09CB: 8D ØE. 99 09DB: 9D 86 86 EB E0 DØ F5<92> 8908:A9 88 BD E4 A9 BD<BB> 02 A<sub>6</sub> 09EB: C4 02 49 RA AD. 38 92 49(9C) SD(4C> 99EB: 66 BD 31 82 A2 28 SD 09F0:48 9D D9 29 EB 63 86 DB<99> 29(09) 09FB:F5 A9 6E **BD** 98 A9 8A88:8D 81 82 A9 CO BD BE. D4<86> 15 9A99: A9 DB(CE) 22 END 2F 92 AD 0A10:C9 25 AD 70 E9(E2) DØ 18 30 28(87) 8A16:88 100 14 49 27 28 95 0A20:01 BD 78 30 EE 59 24 ADC 9F > 49 11<84> PA28:59 2A **C9** 18 DØ 85 0A30:80 59 2A AD 1F C9 07(C1> DØ 8438: D8 05 49 90 BD 79 29 EE(23) 0A40:79 39 AD 79 30 C9 FF D0<3C> 79 30 28 E6(04) BA48: 88 49 8D 66 8A58: 35 AD 39 C9 22 99<08> 78 DØ A9(BE) BA58: AD 1F DØ C9 86 DØ **B**2 20 40 BA60: 91 BD 78 7E 49 48<40> 8468:86 CC AD 88 D4 C9 3D BBC6A> 8478:1F 49 78 8D 99 DA 42 **RRC113** 0A78: A5 CD **D4** 80 19 DB<97> 8D BA. 0A80: E6 CD EB EB ØF DØ F1 A9<3B> 30:88A6 8D 17 DØ 48<13> A6 CC 48 8A98: A9 A4 8D 18 D0 A9 88 BD(48) 0A98: 0A D4 D4 A9 34 BD<27> BD 29 BAAB: 16 DØ A9 BD 17 DØ A9<3B> 16 0AA8: 1E 19 DØ 93 BD 4E 28 B6<10> BARB: 28 B1 20 AF. AF. AC. 20 AB< 1A> 8AB8: A5 A2 AR 8A 88 90 20 @1<3D> 8AC8: 88 86 80 BA 29 01<76> 89 98 9AC8: 99 8A 90 01 88 88 20 B1<BE> 81 8AD8:88 98 88 20A 81 MA B1<88> BADS: 81 81 01 BC BC 710 **BB** BA(2D) 79<70> 0AE.0:04 F3 F3 68 AB 79 90 BAEB: AZ A2 90 70 68 68 79 79(4F) 8AF8: 98 98 B6 86 58 58 79 79(36) BAF8: 98 98 AC. ACK D30 A2 A2 51 51 8998: 98 51 51 88 28 00<AD> 80 51 80 0909:51 98 60 60 08<94> 8919:60 98 5B 98 58 60 00<3D> 88 0818:5B (NA 6C 6C AD. AC. AC: ACC553 79 79 79<5A> 79 0828: 6C 98 79 22 0929:79 88 98 98 70 2969 90 90(13) B6<F7> 0930:90 98 **B6** 86 **B**6 190 86 0838:BA 88 A2 A2 A2 A2 A2 A2<14> 09401A2 20 A2 000 A2 00 **A2** 88KCD> 0B48:A2 88 90 80 70 88 98 88<C4> 085**0:** 90 98 79 606 79 196 79 BB<12> 0858:79 99 6C 6C 6C 4C 6C ACC4C> 9869: 6C 20 68 223 68 20 68 **DB**(FB) 22 98 90 88 90 88(C5) 86:668 20 0B70:90 **5B** 59 28 58 20 59 58<D4> 89 087B: 5B 88 88 68 88 88 80(A3) 0980: 80 26 EA AR AR A2 20 A9(F9) 5C **E**4 89 CB BE<615 **0888:07** 20 BD. 0890: D4 A9 BD **0**8 D2 8D 7D<E52 0998:30 8D 7C 30 BD 30 8D<FB> 78 BRAD: 7B 397 A9 93 BD. 70E D2 68(4B) 85A8: D8 86 38 CE AE. B2<60> **78** BD 0BB0: 2C 63 2D 28 D2<07> ØBBB: AE **7**C 38 RD E5. **2B** CO 00(BD) BBCB: FB DA. BD 86 28 BD **P**11 D2<06) 7C **C**9 A9(19> @BCB: AD 30 07 DØ 29 30 8900:00 BD 7C 80 D2 EE (9E) 08D8:7A 30 EE 7B 301 AD 38(84) 70 MRED: C9 29. DO 69 78499>

Listing zu »Submission« (Fortsetzung)

0868:38 AD 7D 38 C9 98 D0 98(42)	0E90:01 01 01 00 00 3C 3C 24<3B>
0BF0: A9 00 80 7B 30 EE 7D 38<76>	0EB8: 24 24 28 28 28 28 28 28 80(B7)
@BF8:EE 7C 30 AD 7D 38 C9 80<20>	0EC8:08 08 08 00 08 00 08 08 <dc></dc>
0C00:F0 23 AE 78 30 80 05 2C(E0)	0EC8:00 00 00 BE 2F 89 38 E2<5F>
9C99:80 96 02 18 69 91 8D 92<8E>	@ED0:E2 E6 E3 E4 E5 E6 E2 E1<55>
0C18:D2 A9 A4 BD 83 D2 80 07<3A>	0ED8:30 30 30 60 38 90 60 30<45>
0C18:D2 AD 7B 30 C9 80 D6 05 <a3></a3>	BEEB: 96 55 96 55 7C 7C 96 34(42)
0C29:A9 00 8D 7B 30 AE 7A 38<43>	BEEB: DB 58 FB 96 BA 34 94 54<4A>
9C28:BD BE 2B BD 04 D2 C9 00(49)	0EF0:76 D8 08 AS 16 80 32 D2 <e0></e0>
0C30:F0 09 AE 7C 30 BD DE 28<6B>	0EF8:80 40 00 30 00 00 00 00<2C>
0C38:80 05 D2 A9 02 80 08 D2 <d5></d5>	@F 00: 00 90 50 50 50 90 50 00<2E>
8C48: A6 CE 4C 62 E4 A8 62 A2<34>	8F88:88 88 88 88 88 88 88 A8>
0C48:E4 A9 87 28 5C E4 A9 C0<1D>	0F10:00 50 00 00 00 5A 00 00<68>
9C50: 8D 8E D4 A2 80 A9 88 9D<86>	0F18:00 00 00 00 00 50 00 00<77>
0C58:00 D2 E8 E0 09 D0 F8 A9<0C>	0F20:50 00 00 50 00 50 00 50 <f8></f8>
0C60:93 8D 0F D2 60 64 2D 5F(66)	9F28:98 00 00 00 00 00 00 06 58<96>
9C68: 2E 80 80 80 3F 7F FB F0<67>	
	9F38:99 98 98 98 98 58 88
	0F40:00 00 5A 00 00 00 00 00<89>
0C80:1C 00 00 00 00 00 40 70<07>	9F48:5A 00 00 00 50 50 00 46<1C>
0C88:3E 1E 00 00 00 00 00 00<07>	0F50:00 00 00 00 50 00 50 <ff></ff>
0C90:00 00 06 06 1F 2E 30 3E(50)	0F58:00 00 00 00 50 00 00 00 <f0></f0>
0C98:18 18 80 80 18 30 60 60<20>	0F68:58 00 00 00 58 00 08 00 <a0></a0>
OCAM: 30 18 00 00 00 18 14 12<71>	9F48:90 50 90 90 90 90 90 90<9A>
BCA8:12 14 18 10 80 00 00 00 20<02>	0F79:00 00 50 00 00 00 00 00 00<98>
0C90:00 00 00 00 3E 7F FB F0<94>	8F78:50 90 90 E1 90 A0 00 00<4F>
0C88:E0 C0 C0 C0 60 00 1C 3C<5A>	0F88:00 5A 30 00 00 00 00 00 00<38>
OCC0:2E 26 32 78 7C 7E 7E 3C<48>	@F88:90 00 00 00 00 00 00 00 00<
OCC8:19 00 00 00 00 90 81 C7<88>	0F90:00 00 00 00 00 00 00 00 00 AE>
0CD0:E7 66 00 00 00 00 00 00 <f5></f5>	9F98:00 00 00 01 00 1E 1C 01<78>
OCD8: 00 00 06 0A 1F 3E 3B 3E (DE)	0FA0:1A 00 1A 00 90 00 02 04<17>
OCE0:18 18 00 00 10 18 0C 06<89>	9FA8:86 98 98 88 FF 7C 6E 6E <fb></fb>
OCE8:03 01 00 00 28 38 28 2C(5E)	8F88:7C 08 A0 7C A0 A0 A0 B0 <bf></bf>
8CF8:24 64 44 88 88 88 88 88 88 88	@FB8:7C 00 7C 7C 7C 7C 00 7C<19>
OCFB: 00 00 00 00 FC FE 1F 0F(3B)	0FC0:00 7C 00 7C 7C 7C 00 7C <ff></ff>
@D@@:07 @3 @3 @3 @6 @0 3B 3C <cb></cb>	@FCB: 7C @@ 7C BA 30 B5 31 64 <f2></f2>
QDQB:64 78 78 78 78 78 64 7C 3C(34)	0FD8:00 AB 7C 00 AB AB 7C 7C<34>
0010:38 00 00 00 00 00 06 0E<60>	@FD8:A0 7C 7C A0 00 7C 64 64<98>
0D18:7C 78 00 00 00 00 00 00<4E>	@FE8:7C 7C 7C A8 7C 7C 7C 7C47>
0D20:00 00 60 50 F8 74 0C 7C<4C>	9FEB: A0 7C 7C 7C 64 80 7C 7C<42>
0028:18 18 80 00 18 0C 06 06<57>	0FF8:09 7C 7C 80 7C 80 7C 7C<0D>
0030:0C 10 80 00 00 18 28 49(4F>	0FF 8:64 7C 7C 80 7C 80 7C 7C<47>
6D38:48 28 18 88 80 80 80 80 80<	1000:00 7C 7C 64 64 7C 7C 7C<64>
@D60:E7 66 90 90 00 66 2E 58<7C>	
9D48:2F 00 00 00 00 00 40 58<2A>	1838:21 68 88 38 38 21 38 38 92>
@D70:F8 74 @C 7C 18 18 @@ @@<@E>	1838:98 5F 88 38 38 38 38 98 <d5></d5>
0D78:00 10 30 60 C0 80 00 00<0C>	1848:38 88 38 88 38 38 38 86<1D>
0D80:1C 1C 1C 36 22 22 22 02<72>	
@D98:FC FE 3F 1F @F @7 @3 @3<9F>	
0090:06 00 3C 34 26 0E 1E 3E<52>	
@DAG: 1F @F @7 @3 @2 8@ FB 78<95>	1048:38 5F 21 30 30 00 30 30 <a9></a9>
@DAB:00 00 00 00 00 00 00 00<	
8DB8:86 00 00 00 00 00 00 00 00 <ca></ca>	1979:99 39 99 39 39 39 39 39 38(B1)
0DB8:00 00 00 00 40 60 50 F8(EE)	1989:21 60 30 5F 30 30 30 30 <c7></c7>
@DC2:7C @C 7C 18 @2 @8 18 3@<@C>	1888:5F 99 30 30 30 39 SF 30<80>
9DC8:E0 C0 E0 90 98 DC 6C 30<16>	1090:30 60 30 38 38 30 00 60 <b8></b8>
@DDG: 88 88 88 88 88 88 88 86(EA)	1958:59 28 99 69 99 39 29 89 <b9></b9>
2DD8:20 20 20 20 20 20 3F 7F <f0></f0>	1848:48 38 28 48 60 60 40 40 40(13)
@DE0:FC FB F0 E6 C0 C0 60 00 <b2></b2>	19A9: 30 60 05 B1 BE 51 B1 00<39>
@DE8:3C 2C 64 70 78 7C F8 F8<48>	1080:81 B1 51 BE B1 00 B1 00<4A>
9DF0:E0 C0 40 01 1F 1E 00 00 <f3></f3>	1088:81 B1 B1 B1 08 91 00 B1 <fa></fa>
2DF8:28 28 28 28 28 28 28 28 28<12>	19C9:20 B1 B1 B1 00 B1 B1 06<62>
0E00:00 00 00 00 00 00 00 00<1C>	19CB: 81 BE 00 B6 31 B1 32 81(25)
0E88:00 08 08 02 06 0A 1F 3E(18)	1000:81 00 51 BE B1 91 91 B1 <be></be>
ØE18:30 3E 19 00 10 18 0C 07 <d6></d6>	1008:B1 81 98 81 51 BE 81 B1<02>
@E18:03 07 0D 19 38 36 0C 00<75>	1000:81 81 81 81 81 81 81 81 81 81 81
0E20:00 80 80 80 00 00 00 06<3C>	10E9:81 B1 BE S1 B1 B1 00 B1 <f3></f3>
9E28:00 00 00 00 00 00 00 00 44>	10F0:B1 00 81 00 B1 B1 BE B1<18>
0E30:00 00 7E FF FF 7E 7E FF <f7></f7>	19F8:81 86 81 88 81 81 82 91 <ea></ea>
0E3": 7E 24 00 7E 3C 3C 7E 7E(41)	1100:B1 51 BE B1 B1 81 81 81 (E3)
0E40:3C 3C 3C 3C 3C 00 00 00<76>	1108:81 B1 00 81 81 81 B1 B1 <f7></f7>
9E48:48 48 66 68 88 88 9C 9C<8A>	1110:81 81 81 91 81 81 81 80<41>
9E58:08 88 88 88 88 88 88 88 88 6C>	1118:B1 00 81 B1 00 B1 81 81<04>
0E58:00 81 81 00 18 19 19 01(FC)	1129:00 B1 81 00 B1 00 00 B1<15>
9E68:43 C2 88 80 80 SC 2F BD<9E>	1129:01 81 81 2D 2D FF C8 C8(80)
0F68:2F 80 80 00 98 3C 3C 24<89>	1130:C8 C8 C8 FF 20 20 20 20 XB6>
8E78:24 24 84 84 84 84 84 88 <ab></ab>	1138: C8 C8 C8 C8 C8 C8 C8 C8 C8(5A)
8E78:20 00 00 00 00 00 7E FF<90>	1140:C8 C8 C8 C8 C8 C8 C8 C8 C8 C8
0E80:FF 7E 7E FF 7E 24 80 7E(50)	1148:C8 C8 C8 C8 C8 C8 C8 C8 C8(BE)
8E88:3C 3C 7E 7E 3C 3C 3C 3C (EC)	1150:20 20 20 20 20 20 20 20 20 20KF1>
0699:3C 90 00 00 00 00 00 0C 0C4F>	1158:20 20 20 20 20 20 20 20 20 20 20 20
	1158:20 20 20 20 20 20 20 20 20 20 20 20 20 2
0E98:48 40 68 68 88 88 88 88 68 F6>	1168:20 20 20 20 20 20 20 20 20 20 20 52>
9EA0: 20 98 00 00 00 81 81 00 <c5></c5>	
@EAB:98 98 96 80 C2 43 81 01 <c3></c3>	1170:C8 C8 C8 C8 C8 C8 C8 C8 C8 C9<27>

1178:C8 C8 C8 C8 C8 C8 C8 C8(1F> 1189: CB CB CB CB CB CB CB CB(F7) C8 C8 C8 C8 C8 1189:C9 Ca C8(CF> 1190:CB CB CB CB C8 CB CB C8<C6> 1170:CB CB C8 CB CB CB CB(BE) 11AB: CB CB CB C8 C8 C8 C8 **PROCRES** 11AB:CB CB 00 2D 2D 2D 2D 20(18) 1190:00 CB CB C8 C8 2D 20 2D<D4> 1188: 2D 2D 2D 2D 2D 2D 2D 2DKBA> **20** 32 11C0: 2D 20 ΖD 20 2D 2D 2D< A2> 11C8: 2D 2D 2D B2 76 33 2D<17> 1100:CB C8 C8 C8 C8 C8 C8 C8<07> CB(FF) 1108:C8 C8 C8 C8 C8 CB CB 11E0:C8 C8 C8 C8 C8 C8 C8 11E8:20 2D 2D 2D 2D 2D 2D DB(96) 20(09) 11F0:2D 2D 2D 2D 2D 2D 2D 2D<11> 11F8: 2D 2D 2D 2D 2D 2D 2D 20<F9> 1200:2D 20 20 20 2D 2D 2D 2DKDF> 1208:20 20 20 20 20 20 20 20 2D(27) 20 20 2D 2D 2D 20 20<38> 1210:2D 1218:20 20 2D 2D 20 20 2D 2D<17> 1220:20 20 20 20 20 20 20 42<51> 1228:24 81 30 3C 3C 41 A4 89(14) 3C 3C 30 00 00 1A(93> 1230:00 00 1238:06 1A 2E 42 56 6A 7E 92(AF) 1248: 7E 6A 56 42 2E 1A B6 1AKBC> 1248: 1D 89 1D 31 45 59 6D 81<BD> 1250:95 81 6D 59 45 31 1D 89<24> 1258:10 28 8C 28 34 48 SC 70KFD> 1260:84 98 84 70 50 40 34 28KFC> 1268:0C 20 17 83 17 2B 3F 53(F4) 1270:67 7B BF 78 67 23 ØF 2B(EE> 53 SF 1278:17 03 17 23 37 49<BC> 98 87 73 SF 1280:5F 73 87 4B<C9> 0F 23 30 0E 00 1F<FD> 72 08 77 33 35 34<5D> 1288:37 23 PMF 1298:48 50 8D<9C> 1298: A2 88 BE 88 D2 A9 FF 12A0:02 D2 A9 AF 8D 83 D2 A8K36> 12A8:02 20 AØ 33 A9 F3 BD @2<2B> 1289:02 A0 02 20 A0 33 E8 E8<61>
1288:0A D0 E2 A9 00 8D 00 D2<84> 1208:68 84 04 A9 00 85 14 A5KAA> 1208:14 C5 D4 D8 FA 68 F3 79(SB) 12D0:09 6C C1 68 86 58 A2 51<49> 48 A2 BB A9 85(36) 12DR: 90 48 88 12E0:8D 08 D2 BD AD 33 BD 02<20> 12EB: D2 A9 AF BD 83 D2 A8 Ø8<F4> 12F8:29 AB 33 E8 E0 0E D0 EB<B8> 12F9:68 AD 7F-39 C9 81 D8 56(CA) **85**<51> 1388: FE 88 38 AD 88 38 C9 00 80 80 30 EE<E1> 1389: DØ Ø8 A9 1310:81 30 AE 81 39 BD E5 28(89) 1318:80 04 D2 A9 AA SH FD REKEAS 1320-30 ED BB 38 80 85 D2 EC(94) 25 DO 24 A9 7F(93) 1328:7E 22 BD 25 D2 AD 18<37> 1330:30 BD 7E 30 1338:69 08 BD 7E 30 AD 81 30K75> 1348-F9 19 98 8A A9 99 BD B1<83> 1348:39 A7 7E 38 A2 8C(4E) 68 8D 1350:28 87 40 20 B4 3C 68 36(6B) 1358:34 31 35 28 67 34 AE &F (DB) 1369:39 BD EB 2F C9 DO FO 23(67) 95 1368:80 02 DO BC 88 A9(AD) 31 28 1370:FC A2 20 99 7E CB E8<34> BD<07> 23 D2 F7 AE 6F 30 1378:EB 1380:95 31 38 E9 06 A8 A9 30(85) 78 68 A2 08 A9 28<27> 1388:99 86 1390:9D 00 Æ 90 88 78 E8 EØ<48> 1398:FF DØ F3 60 AD ØE DØ C9<9A> 13A0:00 F0 05 A9 01 80 69 3@<3B> 13A8: AD 49 30 C9 01 F0 0E A9</E> 1380:87 BD 06 D0 8D 68 30 A9<23> 13BB: 06 8D C2 02 60 AD 69 38<39> 1308:09 01 F0 01 60 A2 00 CEKF5> 13C8:68 30 EB E0 D6 D0 F8 AD<B6> 13D0:0A D2 8D C2 02 AD 68 30(F1) 13DR: 8D 86 DØ 8D 84 D2 A9 86<D8> 13E0:8D 05 D2 AD 68 315 C9 14<BC> 8D 69 38 80<95> 13E8:80 10 A9 88 13F0:04 D2 BD 105 102 A9 B9 8D<C2> 13F8:68 38 60 AD 1F DØ C9 Ø7(E5) 78 38 ADKB6> 1490 - DO DS A9 20 BD 1408:78 30 C9 92 DØ 28 AD 1F(29) 1418:D8 C9 86 DØ. 21 A9 40 80<CB> 1418:0E D4 A2 00 8E 2F 02 BA<6F> 1428:9D 00 D0 E8 E0 0C D0 F8<F7> 30 20 FD 38<4E> 1428: A9 61 8D 78 1438:20 96 40 40 72 49 68 AD(BF) 1438:84 D8 C9 85 D8 85 A9 81(A6)

Listing zu »Submission« (Fortsetzung)



1440:8D	6B	30	AD	95	DØ	C9	04<17>
1448: DØ	<b>0</b> 5	A9	61	BD	6B	30	C9(F6)
1450:05	D@	25	A9	21	BD	6B	32<5D>
1458:35	E5	35	30	68	AD	<b>6B</b>	30<84>
1468: 09	21	FØ.	01	60	A9	20	85<56>
1468:14	20	52	2D	EE	CØ	02	CE<00>
1470:C1	02	AD	ÇØ	82	8D	66	D2<9F>
147B: A9	CB	ad	61	D2	A5	14	C9<43>
1480:64	DØ	E9	A9	99	BD	80	DØ<78>
1488:8b	61	DØ	CE	7C	48	26	52<20>
							AD <c8></c8>
1490:20	29	BB	22	20	77	33	
1498:70	48	C9	18	00	63	4C	FB <fc></fc>
14A0:34	28	52	20	4C	A6	49	AD<48>
14A8: ØA	DØ	C9	02	DØ	05	A9	01 <e7></e7>
1480:8D	6B	3/8	C9	62	DØ	25	A9<9A>
1486:01	BD	ΔB	26	40	AD	86	30 <e7></e7>
1408:09	01	FØ	22	AD	05	DØ	C9<8F>
1408:07	FØ	24	C9	81	112	17	AD(AF)
1400:70	20	C9	64	90	86	Œ	78 <bc></bc>
1408:30	4C	BE	35	C9	63	BØ	86<71>
14E0: EE	78	38	4C	BE	33	60	28(A5>
14E8: A9	3A	AD	87	30	C9	88	F8<86>
						60	AE <es></es>
1460:05	A9	1F	80	68	38		
14F8:6F	20	BD	DD	46	C6	90	DB <ae></ae>
1500:00	A2	19	20	87	48	ΑE	6F<3D>
1508:38	A9	81	9D	DD	40	60	E6<65>
1510:35	E1	36	EE	60	39	AD	6D <a5></a5>
			-				
1518:30	C9	14	DØ	17	A9	1912	BD <f8></f8>
1520:6D	30	<b>GD</b>	6E	30	20	22	3F <c8></c8>
1528: 20	87	3E	20	45	3F	20	B9 <ca></ca>
1530: 3F	26	DF	3F	C9	8A	DØ	89 <c8></c8>
						A9	01<01>
1538: 20	45	2E	20	89	3F		
1540:8D	9E	30	A9	CE	18	6D	6E <c2></c2>
1548: 30	BD	1B	23	ap	16	25	8D<9D>
1550:D9	27	68	AE	6F	30	A9	88<10>
1558: 9D	2F	02	A9	20	ap	39	02<84>
1568: A9	69	ap	31	92	BD	22	28<3B>
1568:85	58	BD	B2	28	82	59	A8 <d2></d2>
1570:00	B1	58	99	126	26	CB	CB <d2></d2>
1578:50	DØ	F6	8D	98	30	gp	03<43>
		79	36	28	36	34	20<6F>
1580: De	20						
1588: CE	32	20	80	37	20	2D	37<70>
1590: A9	3E	ab	2F	92	68	A9	00<02>
1598:85	14	A9	88	8D	1E	DB	A5 <se></se>
15AB: 14	C9	14	00	F5	66	A2	98 <f2></f2>
15A8: A9	90	90	99	7F	EB	EØ	FF<35>
1580: D0	F6	AE,	6F	30	BD	18	31 <e2></e2>
1599: 8D	93	20	BD	95	31	80	94 <c6></c6>
1500:30	AE	93	38	A9	<b>C</b> 3	9D	88<11>
15C817F	EB	EC	94	30	DØ	F5	AEK3F>
		_					
1500:93	26	A9	FF	9D	98	7F	9D <d6></d6>
1508:01	7F	BA	18	69	ØA	AA	ECKE1>
15E0:94	20	99	EE	68	A2	60	ACKBA>
15EB:71	30	BD	F4	20	99	88	7C <f5></f5>
1,5FØ: BD	18	ZE,	79	22	70	CB	E8 <a9></a9>
15F8:E0		De	EE	60	A2	20	AC<29>
	24						
1600:71			64	20	99	90	70<12>
1600: BD	88	20	99	98	7D	C8	E2<47>
1610:36	47	37	E9	60	24	DØ	EE(BD)
1618:60	AZ	(96)	AC	71	38	BD	30<91>
			7C	80		2E	99(AZ>
1620:2E	99	98			68		
1628:00	70	CB	83	EØ	23	200	EE<44>
1630:60	A2	98	AC	71	30	BD	ACKE9>
1638: 2D	99	20	70	BD	DØ	2D	99(89)
1640:00		CB	EB	EØ	23	DØ	EE<1C>
							C9(1F)
1648:60	EE	72	30	AD	72	30	
1650:02	DØ	85	A9	98	80	72	30 <f7></f7>
1658: A9	60	BD	73	30	40	EE	77<63>
1660:38	AĐ	77	30	C9	02	DØ	BB<88>
1668:A9		BD	62	82	A9	96	
1670:77	30	90	A9	36	BD	C6	02(80)
1678:60	48	37	43	38	AZ	88	90<07>
1680:FB	80	EB	EØ	88	DØ	FΒ	69<52>
1689: AZ	96	9D	98	81	E8	EØ	@B <ac></ac>
1690: DØ	FB	60	EE	83	30	AD	83<40>
1698: 30		32	DØ	BA	A9	18	20<95>
16AØ: 4B	37	A9	90	28	22	37	C9<59>
1648:64	DØ	ENG.	A9	80	20	48	37 <ce></ce>
1680:A9			53	37	C9	96	DØ<38>
						20	53(54)
1488:08	A9	99	8D	83	30		
1600:37	60	20	48	38	Æ	6F	30<5E>
16CB: BD	DD	40	C9	21	DØ	13	AD<7E>
1600:6A		80	89	25	8D	98	
1608:38		01	8D	66	23	BD	6D<89>
1660:23		C9	02	DØ	15	A9	88<13>
16E8:8D		25	8D	66	23	AD	6A<1F>
16F0: 30	BD	98	25	28	E9	21	80<90>
16F8: 6D		60	C9	03	DØ	15	A9<6A>
1700:00		78	25	8D	6D	23	AD <a2></a2>
1708: 6A			89		38	E9	01<6E>
1710:8D	66	23	60	C9	24	DØ	@E<74>

1718:A9 00 8D 89 25 8D 90 25<56> 1720:80 23 8D 6D 23 60 AD(CF) 1728:05 D0 C9 08 D0 34 A9 @1<@8> 30 C9 78<5A> 173B: 80 7F 30 AD 70 90 8D<01> A9 25 1738:90 11 99 BD 1740:60 23 AE 03 9D(E1) 6F 38 1749: DD 40 68 AD 70 30 09 6E(7A) 8D 89 25 RD(CB) 1750:80 18 A9 00 1758: 66 23 AE **AF** 30 49 82 9D<000> 98 69 DØ<8A> 1760: DD 4A AD 23 88 DØK 440 1768:10 AD 90 25 CD 66 23 1776:08 A9 84 **AE** 6F 30 DD<783 177B: 4A 44 38 CC 38 AD EB 4A(14) FB 75 BD<D4> 1780: 8D 48 68 AE 30 1788:8E 2F BD 6A 30 69 BE. 76<0A> 1790:30 AD 76 301 09 85 DRI RIDK DCC 1798: A2 00 BD BA 33 9D 81<78> 10 DØ F5 C9 ØA DØKAB> 17A0: E8 EØ 28 A2 8E 30 BD 12(10) 17A8:10 00 76 9D DØ<38> 17B0:33 10 81 EB EØ 86 17981F5 60 EE 82 38 AD 82 38(C2) 1700:C9 08 DØ 1C 49 90 SD 27(FE) 17C8: 26 80 2E 26 an FE 27 80×4C> 28 49 9D 28 8D<6B) 1700:06 BD 26 1708:31 26 BD FF 27 80 97 28<6B> 18 21 A9 9D RID 27(88) 17E8: C9 DØ 8D<AC> 17EB: 26 8D ZE 26 an FE 27 AP 2A BD(FB) 28 9C 26 17F0:06 ŒΦ 17F8:31 26 BD FF 27 BD 87 28<4B> 1900:A9 00 BD 92 30 38(2E) 60 CD 1808: CB 39 4B 18 AD PIR 134 C9(F7) 1818:46 BD 1.0 AD 74 30 BD BA<81> 1819:D4 BD 18 DØ A9 88 B9(FC) BD **A9** 1920: D4 48 48 78 BD BA D4(7C) 1828:8D 09 D4 AP PR AD 18 00(38) 1830: BD 1A DØ A9 18 80 17 DØ<82> 1838:48 40 99 Δ9 22 99(E6) **AZ** 90 1840:70 90 00 7D 9D 20 7E 9DKAC 78 EB **EF** 60<83> 1848: 00 EB DØ ED 1950:EE 9A 38 AD 90 38 C9 03<3D> 49 20 67 20 39(40) 1858: DØ 39 AB 09 39 20 3A 89 88<C2> 1860: 20 12 30 AC 95<02> 1848: 8D 9A 30 EE 95 6043F 1870:30 BE 18 33 A9 A7 18 1878:08 20 90 37 26 9D E1 26<FC> BE 1.B A9 88 9D 37<58> 1680:CB 33 1888: 26 90 26 88 88 18<90> E1 BE 9D 37 26 AD<34> 1090:33 9D E1 26 1898: 95 38 C9 ØF DØ 85 49 88<15> 18AB: 8D 95 3/8 60 EE 96 30 AC<42> 18A8: 96 20 BE 2C 33 A9 **A7** 18(CD) 1880: 6D BB 30 9D 37 26 9D E1<34> 1998: 26 CS BE ZC 33 A9 DUD 90(41) 26 68 1800:37 26 88 BE(51) 9D E1 1809: 20 33 9D 37 26 9D 26(DD) E1 1800: AD 96 32 C9 8F DØ 65 A9(69) 18D8:00 60 9D 96 38 97 38<4D> EE IBE 0: AC 97 TO DE 330 33 69 A7(A9) 37 9D<7E> 18E8: 18 6D BB. 38 90 26 33 26 BE 00(FE) 19F0:E1 CB 3D 49 18F8: 9D 37 26 9D E1 26 88 88(48) 39(AD) 1700: BE 3D 300 9D 37 26 **C9** 1900: 4A 30 90 E1 26 AD 97 30<DC> 97(AA) 1910:C9 ØF DØ 85 **A9** 20 BD 1919:30 69 78 30 AC 98 38(41) EE 33 A9 1920: BE 4E A7 18 6D 8B<70> 1928:36 90 37 24 en. Fit 26 CB<47> 33 69 88 1930:BE 45 9D 37 26KEC) 1938:90 E1 26 88 BE 4E 33(3F) 88 1940:9D 37 26 9D E1 26 AD C9 @F DØ @5 00 BD(7A) 1940:30 **A9** 1950:9B 301 40 EE 99 30 AC. 99(18) 5F 1959:30 BE 33 A9 A7 18 6D<A6> 9D 37 9D 26<1A> 1960:8B 30 26 El 1948 + C8 BE 5F 33 A9 9D 37(43) 1970:26 9D E1 26 88 88 BE 5F (98) 1979:33 9h 37 26 90 E1 26 ADCSA5 C9 ØF A9 1980:99 30 DØ 05 00(39) 1989: 8D 99 30 60 **4B** 38K4E) 46 73 30 AD 1990: EE 73 30 **C9** 08(15) 1998:00 03 20 1B 37 49 PR RD(F2) 19A0: B5 68 AD B6 30 C9 20(76) 30 1968: DØ 43 AD 78 92 C9 27 DB(B5) 1980: 18 EE 70 30 20 **A9** 3A ADC443 1988:72 30 C9 88 DØ ØR. 00 DAKAAS 19001 AD 60 301 4C DØ 34 40 90(CE) 1908:37 C9 0B D0 CE 70 30(AF) 18 1900:20 A9 3A AD 30 C9 72 4C(84) 1908:D0 08 A9 A4 BD 6C 30 19E0:88 36 4C EB 36 AD 87 38(87) 1968: C9 01 D0 01 EA 60 AD 70KE5>

AE<1B> 19F8-38 80 88 08 80 81 08 89 EE(7F) 30 DD 12 32 D0 19FB: 6F 4C 3C(8B) 1A00: 6F 30 20 36 26 1AØ8:DD BE 32 DB 89 ÇE 6F 30(DE) 60 1A10:20 26 4C 1019 3C Pb(B2> AD RF 1A18: 20 34 DØ **Ľ**9 DOK(C4) 77 1A20:FB @2<9B> 50 29 **7B** 3B AD 78 1A28: C9 0D D0 30 C9<E4> AD 85 12 A9 ØØKBE> 71 30 1A30:01 F0 ØB EE 1A38:8D 85 38 20 33 **3B** 28 C6<68> 12<3C> 1A40:38 AD 78 02 C7 0E DØ 30 C9 82 FB ØB. CE(70> 1A48: AD 85 38 20<35> 1A50:71 49 **02 AD 85** 30 DR(4B) 1A58: 33 3B AD 85 38 09 88 30 BD 99 30 SD<BD> 1A60:11 AE 6F A9 88 BDK7CX 1A6B: 78 30 20 A9 3A 1A70:85 38 A9 88 BD 16 DØ 68(6D) 1A78: AD 72 30 C9 00 D0 63 4C(A3) 00(22) 5E 3B A2 1ABØ: 41 **3B** 4C 69 1A88: AC 71 30 BD 47 3B 42 3E(A6) 99 2F<67> 1A98:26 2F 20 7C BD 44 1A9B: 99 98 7D EB CB 24 DØKFD> EØ 1AA8: EE A9 81 8D 86 30 40 A2<7A> 1AAB: 00 AC 71 30 BD δE 2F 99(93) 1AB0:00 7C RD 92 2F 99 88 7D(ED) E9 24 DØ EE. A9 81(99) 1ABB: CB E8 1AC0:80 30 68 AΕ δF 30 AD(C7) 88 30 18 69 21 95 31<71> 1ACB: 71 DD 1ADØ: DØ 30 A9 81 SD 30 85 A9(2C) 1ADB: 00 8D 86 38 AE 6F 38 RD(AB) **C9** DB 6F<92> 1AE0: 95 31 BE 28 EE 30 **C9** FP 0A<4D> 1AEB: 30 AD 6F 3D 1AFB: C9 29 98 86 Œ 6F 30 CE(93) 1AFB: 6F 38 20 FD 38 20 26 36<BB) 1800: AE 6F 38 BD 18 31 69 01<10> AD<4B> 1908: OD 71 30 28 41 **3B** 60 1810:71 30 AE 6F 30 DD 18 31(05) 1910: DO 39 A9 02 8D 85 30 A9<21> 1820:00 BD 86 3B AE 6F 39 ADCR15 C9 DØ CE 1B20:18 31 21 25 AF(EF) 6F 30 C9 1830:30 AD 3C FB @AKBB) 1838:C9 07 99 86 EE 36 EE(5E) 1848:6F 20 26 36 BD 95 31(10) 30 1B48: E9 23 BD 71 30 28 FD 38(88) 18 1950:20 SE 39 68 AD 70 30<23> 1858: 69 70 90 09(04) 96 BD 30 85 1960:30 8D 78 30 AD 70 38 80KE5> 1868: 00 00 8D W1 60 AD DØ 88(13) 1870: 38 AD 88 30 ED 28 D2 A9<88> 1878: A4 8D 01 D2 60 A9 28 BD<AB> 1888:09 D2 AD 89 26 8D 01 D2<E1> 1888: 40 AD 6C 30 43 3C 39 3D<43> 18901C9 AN FO 11 CE 6C 3.68 CE < 78> 1899: AC 30 A9 CB BD 06 D2 ADKE 70 1BA0: 6C SD AD 85<89> 30 97 02 68 20 19A8: D0 C9 82 DØ 32 52 20(52) AB 00 1990:20 85 20 **A2** FA 20(32) 1988: 87 48 C8 CB BA DO FA ADC SEC 48 15 FØ 7C<E42 1BC0:7C C9 03 EE 70 **C9** 18C8: 48 AD FØ F9(97) 388 98 1800:20 44 2D 20 **C7 3C** EE 75(46) 86 1808: 30 EE 72 48 4C 49 ABCAES I RED : AE 75 30 FA 07 DO NIS A9(D5) 75 1BEB: 00 80 269 AA A9 11 BD(85) 1BFØ: 72 48 BD 6F 33 BD 30(D4) 33 A9 19F8: BC 6F 01 99 DD 4A<4B> 1000:40 EE 6C 48 AD 4C ΔR 09(62) 1008: IA DO 08 EE AB. 48 69 10KFE> 101018D 60 48 60 EA CE &C 4B(62) 1C18: AD 6C 48 E9 WE DØ @F AD(99) 1020:68 48 C9 10 FA 69(1E) 37 CE 1C2R: 48 A9 19 BD 6C 48 A2 FF(31) 8A E9 BC(99) 1C38:A9 00 14 38 85 DØ 1C38: AA A5 14 C9 01 FA BE<972 1C40:00 D2 84 18 69 01 ED 82(22) 1C4B: D2 A9 A8 BD 01 D2 9D M3<3D> 1050:02 FØ AB BB DB **A2** 32 28(88) 9 1059:97 4C 3C 00 BD<6F> 40 C7 1C60:00 D2 BD 101 02 BĎ D2<69> 1068:80 83 60 AD 38(D1) **D2** ØA D2 1070:08 A9 90 SD DB 30 09 A4(5B) 89 1263 80 EE<4D> 1C78:80 EE 38 68 1CB2: 30 Δ9 64 BD DB 38 68 SAKED) 1C88:3D 35 3E AD 89 30 C9 88KE33 1098:00 30 AD 87 38 **C9** DIG DØ(42) 1C98:35 AD 86 3Ø C9 68 DIS 2E<5E> 1CA0: AD 84 82 C9 1200 27 DØ AD(27) 1CA8: 78 **Ø2** C9 **MB** ne Ø1<D7> 1CB0:00 87 38 4C 83 3D **C9** 87(41) 1CB8: DØ 88 A9 82 80 87 30 40<36> DØ 88 A9 1000:83 3D 09 ØF @3<46>

Listing zu »Submission« (Fortsetzung)

4000 00 07 To 40 00 00 40 00 00		
1CC8:8D 87 38 4C 83 3D A9 20(C3)	1FA0:1A DO 08 EE 61 48 A9 10(35>	2278:46 5C 22 06 46 D4 22 06<67>
1CD0:8D 07 30 60 A9 00 8D 88 <a5></a5>	1FAB:8D 62 4B AD 61 4B C9 1A<31>	2280:46 FC 22 06 46 5C 22 86 <f9></f9>
1CD8:30 CE 71 30 CE 71 30 CE<0B>	1FB0: D0 08 EE 60 48 A9 10 8D(2E)	2288:42 2E 48 02 02 41 F0 42(89)
1CE0:71 30 CE 71 30 68 AD 87(8D)	1FB8:61 48 AD 60 48 C9 1A D0<2E>	2290:78 78 C6 8C 22 46 5C 22<23>
1CEB: 30 C9 01 D0 06 CE 70 30 <ff></ff>		
	1FC0:08 EE 5F 48 A9 18 8D 60<9B>	2298:86 46 EC 23 86 46 EC 23<55>
1CF0:4C A9 3D C9 02 D0 03 EE<20>	1FC8:48 AD 5F 48 C9 1A DØ Ø8 <f9></f9>	22A0:06 46 7F 24 06 46 7F 24 <f3></f3>
1CF8:70 30 20 A9 3A 20 9E 3E <ce></ce>	1FD0: EE SE 48 A9 10 8D 5F 48 <cd></cd>	22A8:06 46 7F 24 06 46 7F 24<00>
1000:AD 97 30 C9 01 D0 2E AD<55>	1FD8:AD 5E 48 C9 1A D0 08 EE(E7)	2280:06 46 4C 23 06 46 74 23<5A>
1000:88 30 C9 10 80 03 4C MATONIA		
	1FE0:50 48 A9 10 8D 5E 48 60(C3>	2288:06 46 5C 22 86 42 2E 48 <d4></d4>
1D10:3E C9 1E B0 03 4C CA 3D<74>	1FE8:8E 84 30 A2 00 20 38 40(AF)	22C0:02 02 41 23 43 70 TM C6(5B)
1D18:4C CA 3D AE 71 30 A0 00<01>	1FF0:E8 EC 84 30 00 F7 60 A2<45>	22C8: 0C 22 46 5C 22 06 46 5C(1A)
1D20:89 84 2E 9D 80 7C 89 AE(34)	1FF8:FF E8 BD 5D 48 C9 10 F0<92>	2200: 22 86 46 5C 22 86 46 5C<36>
1D28:2E 9D 00 7D E8 C8 C0 24<82>	2000:F8 8E 93 30 A2 FF E8 RD(BA)	2208:22 06 46 5C 22 06 46 37(29)
1D301D0 EE 4C 12 3E AD 88 30<5C>	2008:EF 29 C9 10 F0 F8 66 94 <f2></f2>	22E0:26 06 06 06 06 06 06 06 06 34>
1D38:C9 10 B0 03 4C 6A 3E C9<4E>	2010:30 AD 93 30 18 CD 94 30<34>	22E8: 46 SC 22 86 42 ZE 48 82<5E>
1D40: 1E B8 83 4C F8 3D 4C F8(9F)	2018:80 0F AE 93 30 BD 50 48<4C>	22F0:02 41 58 43 70 70 C6 BC<59>
1D48:3D AE 71 30 A0 00 89 DZ <fe></fe>	2020:90 EF 29 EB E0 06 D0 F5(89)	22FB: 22 46 5C 22 06 46 0C 22(EF)
1050:2E 90 00 7C 89 FD 2E 90<38>	2028:60 DO 18 AE 93 30 BD EF<10>	2300:06 46 AC 22 06 46 5C 22<94>
1058:00 7D E0 C0 C0 24 D0 EE<87>	2030:29 18 DD 5D 48 90 E3 D0<12>	2308:06 46 5C 22 06 46 5C 22(AZ)
1D60:4C 12 3E EE 88 30 20 25<4E>	2038:00 EE 93 30 AD 93 30 C9(AC)	2310:06 46 5C 22 06 46 5C 22<8A>
1D68: JC 20 94 35 AD 80 30 C9(5F>	2040:08 F0 03 4C CA 40 60 E6(97>	2318:06 46 5C 72 06 46 5C 22(82)
1D78:1E 90 36 A9 00 80 00 D2 <e3></e3>	2048:40 E1 41 70 70 C6 0C 22(D4)	2320:86 42 2E 48 02 02 41 87<15>
1D78:8D 1E D0 A9 10 0D 89 30<5B>	2050:46 5C 22 06 46 EC 23 06<55>	2328:43 70 70 C6 0C 22 46 5C(E7)
1D80:FE 71 30 EE 71 30 EE 36<10>	2058:46 EC 23 06 46 5C 22 06(69)	2330:22 B6 46 6F 25 86 86 86<9B>
1088: 3E B6 3E 71 30 EE 71 30(78)	2060:46 D4 22 06 46 FC 22 06(08)	2338:46 5C 22 86 46 5C 22 86 <c6></c6>
1090: AD B7 30 C9 01 D0 08 A9(5B)	2068:46 5C 22 06 46 0C 22 06<2A>	2340:46 5C 22 86 46 5C 22 DA(E3)
1D98:00 BD 87 30 20 9E 3E 4C<49>	2070:46 34 22 06 46 5C 22 86(EF)	2348:43 D5 44 Ø6 46 A7 24 Ø6<65>
1DA0:E8 36 A9 00 8D 87 30 20<7A>	2078:42 2E 48 02 02 41 E6 40(AF)	2350:46 A7 24 06 46 5C 22 86<84>
1DA8:9E 3E 4C 20 37 60 AD 89 <a0></a0>	2080:70 70 C6 8C 22 46 5C 22<4F>	2358:42 2E 48 02 02 41 BC 43<03>
1080:30 C9 00 F0 09 CE 89 30(88)	2088:06 46 BC 22 06 46 34 22(C4)	2360:70 70 C6 0C 22 46 5C 22<79>
1DBB:CE 89 30 20 34 3C 40 AE<25>		
	2090:06 46 5C 22 06 46 0D 28(9A)	2368:86 46 5C 22 86 46 5C 22 <c2></c2>
1DC0:71 30 A0 1D B9 84 2E 9D(D2)	2098:06 46 0D 2B 06 46 9C 23<25>	2370:06 46 5C 22 06 46 24 23<6D>
1DC8:00 7C B9 AE 2E 9D 00 7D(9B)	20A0:06 46 4C 23 86 46 74 23(76)	2378:86 46 24 23 86 46 5C 22(2C)
1DD0:E8 88 C8 00 D0 EE 4C 12(61)	20AB: 06 46 5C 22 86 42 2E 48(E8)	2380106 46 CF 24 06 46 F7 24(CE)
1DDB: 3E AE 71 30 A0 10 89 D2<62>	2000:02 02 41 18 41 70 70 C6<8A>	2388186 46 5C 22 86 42 2E 48(E2)
1DE0: ZE 9D 00 7C 89 FD 2E 9D <c9></c9>	20BB: 0C 22 46 5C 22 06 46 4C(D7)	2390:02 02 41 EF 43 70 70 C6 <d8></d8>
1DEB: 00 7D E8 88 C0 00 D0 EE<62>	20E0:23 06 46 74 23 06 46 9C(6C)	2398:0C 22 46 5C 22 06 46 24 <d5></d5>
1DF0:4C 12 3E AE 71 30 CA CA<19>	20008:23 06 46 90 23 06 46 90(E7)	23A0:23 06 46 74 23 06 46 5C<03>
1DF8:CA CA CA AD 88 A9 88 9D<18>	2000:23 86 46 90 23 86 46 90(FF)	23A8: 22 86 46 EC 23 86 46 14<98>
1E00:00 7C 9D 00 7D EB CB C0<83>	2008: 23 86 46 9C 23 86 46 9C(17>	2380: 24 86 46 5C 22 86 46 5C(AB)
	20E0: 23 86 42 2E 48 02 02 41(C5)	
1E08:32 D0 F4 60 B7 3E B2 3F<14>		2388:22 Ø6 46 SC 22 Ø6 46 SC <a9></a9>
1E10:AD BC 30 C9 18 D0 05 A9(CB)	20E8:50 41 70 76 C6 0C 22 46<8A>	23C0:22 86 42 2E 48 92 02 41<53>
1E 18:00 00 8A 30 C9 21 00 05(58)	28F8:9C 23 86 46 24 23 86 46<28>	23C8:24 44 70 70 C6 0C 22 46<1E>
1E20:A9 01 8D 8A 30 AD 8A 30 <a6></a6>	20F8: 24 Z3 06 46 5C 22 06 46 <f5></f5>	23D0:5C 22 06 46 5C 22 06 46<9F>
1E28:C9 00 F0 31 CE 8C 30 AE<88>	2100:00 28 06 46 00 28 06 46<48>	23D8:E1 26 06 06 06 06 06 06<93>
1E38:BC 38 A9 98 90 34 22 A9 <ff></ff>	2108:9C 23 86 46 24 23 86 46(85)	23E0:06 46 5C 22 06 46 5C 22<6A>
1E38:91 9D 35 22 9D 36(A9)	2119:74 23 06 46 5C 22 86 42(B2>	WWW 46 SC 22 86 46 SC 22<42>
1E40:22 A9 00 9D 37 22 AD 8B<67>	2118:2E 48 82 02 41 85 41 70<84>	23F0:86 42 2E 48 02 02 41 59 <f7></f7>
1E48:30 C9 81 D0 0F A9 93 9D(CD)	2120:78 C6 8C 22 46 5C 22 86<15>	23F8144 70 70 C6 QC 22 46 5C<96>
1E50:34 22 A9 95 9D 36 22 A9<48>	2128:46 31 27 86 86 86 46 5C(CF)	06 46 4C 23 06 46 74<78>
1E58:94 9D 35 22 60 EE 8C 30 <c8></c8>	2138:22 86 46 31 27 86 86 86<83>	2408:23 06 46 9C 23 06 46 1F <c2></c2>
1E60: AE BC 30 A9 96 9D 34 22<14>	2138:46 SC 22 86 46 31 27 96<22>	2410125 06 46 5C 22 06 46 9C(4C)
1E68: A9 94 9D 35 22 A9 99 9D <c7></c7>	2140:06 06 46 SC 22 86 42 E2(13)	2418:23 86 46 EC 23 86 46 14(CC)
1E70:36 22 A9 00 9D 33 22 AD(00)	2148:41 DD 42 2E 48 02 02 41<1C>	2428: 24 86 46 5C 22 86 42 2E<42>
1E78:88 38 C9 88 D8 8F A9 97<24>	2150:BA 41 70 70 C6 BC 22 46(ED)	2428:48 02 02 41 88 44 70 70 <bf></bf>
1E80:9D 34 22 A9 9B 9D 36 22 <a4></a4>	2158:5C 22 06 46 C4 23 06 46<73>	2430:C6 0C 22 46 5C 22 06 46<9C>
1E88:A9 91 90 35 22 60 EE HUCHO	2160:C4 23 86 46 7F 24 86 46<48>	22 86 46 84 22 86 46(1A)
1E90:30 AD 8B 30 C9 02 D0 05<57>	2168:EC 23 06 46 14 24 06 46<11>	2448:5C 22 06 46 5C 22 06 D6 <a1></a1>
1E98: A9 00 8D 88 30 60 AD 8E<6F>	2170:5C 22 86 46 8C 22 86 46 <c9></c9>	2448:44 D1 45 46 1F 25 86 46(C3)
1EA0:30 C9 01 D0 03 CE 80 30 <e0></e0>	2178:34 22 06 46 5C 22 86 42<2A>	2450:50 22 06 46 6F 25 06 06<92>
1EA8:C9 00 D0 03 EE BD 30 AD<22>	2180:2E 48 02 02 41 E9 41 70(EA)	DERMIN 46 5C 22 86 42 2E 48(A2)
1EB0:8D 30 C9 1A D0 05 A9 00 <df></df>	2188:70 C6 0C 22 46 5C 22 86<90>	2460:02 02 41 BD 44 70 70 C6 <b0></b0>
1EB8:8D 8E 30 C9 23 D0 05 A9(1C)	2190:46 4C 23 06 46 74 23 06(D9)	2468:0C 22 46 5C 22 06 46 C4<27>
1EC0:01 BD BE 30 AE BD 30 A9(1E)	219B: 46 9C 23 86 46 9C 23 86(96)	2478:23 86 46 C4 23 46 5C(EA)
1ECB: 9A 18 6D 8B 30 9D 74 23(9B)	21A0:46 9C 23 86 46 9C 23 86<5E>	2478: 22 86 46 6F 25 86 86 86<56>
1ED0:9D 43 25 A9 00 9D 44 10 AM	21AB: 46 EC 23 B6 46 14 24 06<1C>	2480; 46 5C 22 86 46 D4 ZZ 86(28)
1ED8:9D 42 25 9D 75 23 9D 73 <c8></c8>	2180:46 5C 22 86 42 2E 48 02<82>	2488:46 FC 22 86 46 5C 22 86(FD)
IEE0: 23 68 AD 90 30 C9 81 D8 <aa></aa>	2188:02 41 1E 42 70 70 C6 0C(8C>	2490:42 ZE 48 02 02 41 F0 44(A7)
1EEB: 03 CE 8F 30 C9 80 D0 03(D0)		2498: 70 70 C6 8C 22 46 SC 22<2F>
	21C0:22 46 9C 23 86 46 EC 23 <ae></ae>	
1EF0:EE BF 30 AD 8F 30 C9 18<76>	21CB:06 46 14 24 06 46 5C 22<63>	24A0:06 46 0C 22 06 46 E5 27<4B>
1EF8: DB 85 A9 80 8D WM 30 C9<46>	21D0:06 46 5C 22 06 46 5C 22<76>	24A8:06 46 5C 22 06 46 6F 25<3C>
1F00:21 D0 05 A9 01 BD 90 30(89)	2108:86 46 5C 22 86 46 5C 22<7E>	2480:06 06 06 46 5C 22 06 46<4E>
1F08: AE BF 30 A9 93 3F 37 40<78>	21E0:06 46 5C 22 06 46 5C 22(66)	2488:00 22 06 46 B4 22 06 46(9E)
		24C0:5C 22 86 42 2E 48 02 02<1C>
1F10:9C 18 6D 8B 30 9D 14 24<88>	21E8:86 42 2E 48 02 02 41 53<15>	
1F18:9D ØD 25 9D AC 22 A9 9D(28)	21F8:42 70 70 C6 8C 22 46 5C(A8)	24C8:41 23 45 70 70 C6 0C 22(E1)
1F20:38 ED 88 30 9D AD 22 A9(96)	21F8:22 86 46 C4 23 86 46 C4<53>	24D0:46 50 22 86 46 80 22 86<59>
1F28:00 9D 0C 25 9D 0E 25 9D<5D>	2200:23 86 46 9C 23 86 46 9C(23)	24D8: 46 84 22 86 46 5C 22 86(F2)
	2208:23 06 46 1F 25 06 06 06(4D)	24E0:46 6F 25 86 86 86 46 5C(5D)
1F3D: 15 24 9D 13 24 9D A0 22 <b3></b3>		
1F38:9D AE 22 60 AD 92 30 C9<93>	2210:46 9C 23 06 46 9C 23 06 <f3></f3>	24E8: 22 06 46 0C 22 06 46 E5(E5)
1F40:01 D0 03 CF 91 30 C9 00<0F>	2218:46 9C 23 86 42 2E 48 82<7E>	24F0:27 06 46 5C 22 86 42 2E(89)
AFME DE 03 EE 91 30 AD 91 30KFD:	2220:02 41 88 42 70 70 C6 0C <a5></a5>	24F8:48 02 02 41 56 45 76 70<4D>
1F50:C9 18 D0 05 A9 00 8D 92(B4)	2228:22 46 5C 22 86 46 24 23<2F>	2500: C4 0C 22 46 5C 22 86 46(92)
1F58:38 C9 21 D8 85 A9 61 8D<5B>	2230:06 46 24 23 06 46 7F 24 <e7></e7>	2508: 6F 25 86 86 96 46 5C 22<17>
	2238: 06 46 7F 24 06 46 7F 24<4B>	2519:06 46 0C 22 06 46 AC 22(55)
1F60:92 30 AE 91 30 A9 90 18(13)		2518:06 46 5C 2Z 06 46 BF 25<70>
1F68:6D 88 30 60 88 30 60 88(55)	2240:06 46 7F 24 06 46 EC DE<26>	
1F70:30 9D 84 22 A9 91 18 6D <fd></fd>	2248:42 D9 43 23 06 46 14 24<13>	2528:86 46 BF 25 86 46 5C 22<40>
1F78:88 38 6D 88 38 6D 88 38(22)	2250:06 46 5C 22 86 42 2E 48 <ac></ac>	2528:86 42 2E 48 02 02 41 89(F1)
1F80:90 85 22 A9 98 18 60 8B <d0></d0>	2258:02 02 41 BB 42 70 70 C6(46)	2538:45 78 78 C6 8C 22 46 SC(D2)
1F88:30 9D 86 22 A9 00 9D 87<08>	2260:3C 24 46 5C 22 86 46 24<32>	2538: 22 06 46 4C 23 06 46 Z4<72>
	2268: 23 Ø6 46 24 23 Ø6 46 5C <cz></cz>	2540: 23 06 46 5C 22 06 46 D2<93>
1F90:22 9D 83 22 60 38 40 E5(E2)	ZZ00:Z0 00 70 Z4 Z0 00 40 DL\LZ/	
	DOME NO AL AL LE DE DI DI DICE	
1F98:48 EE 62 48 AD 62 48 C9<7B>	2270:22 06 46 6F 25 06 06 06<52>	2548:45 CD 46 EC 23 86 46 14(C9)
1F98:40 EE 62 48 AD 62 48 C9<78>	2270:22 06 46 6F 25 06 06 06<52> Listing zu »Submission« (Fortsetzung)	2548:45 CD 46 EC 23 86 46 14(L9)

								_
2550:24	86	46	50	22	86	46	2C<7C>	
2558: 22	106	46	34	22	86	46	5C<42>	
2568: 22	86	42	2E	48	02	82	41 <b7></b7>	
256B: BC	45	76	70	C6	ØC	22	46 <cf></cf>	
2570:5C	22	06	46	EC	23	86	46 <d4></d4>	
2578: <b>9</b> F	26	66	46	50	22	86	46 <b2></b2>	
2580:40	23	26	46	24	23	96	46(DA)	
2589:5C	22	86	46	ØC.	22	86	46<69>	
2590:34	22	196	46	5C	22	86	42<48>	
2598:2E	48	02	02	41	Fī	45	78KF2>	
25A0:70	C6	ØC.	22	46	50	22	Ø6<9D>	
25AB: 46	@D	28	96	46	GD.	29	Ø6 <f2></f2>	
2580:46	SC	22	86	46	5C	22	66<43>	
25B8:46	50	22	86	46	5C	22	86 <aa></aa>	
2500:46	4C	23	26	46	74	23	06<11>	
2508:46	50	22	86	42	2E	48	@2<14>	
	41							
2500:02		26	46	710	70	C6	ØC<69>	
25D8: 22	46	5C	22	96	46	E7	25<6E>	
2560:06	46	E7	25	06	46	<b>5</b> C	22<11>	
2568:86	46	CF-	24	66	46	F7	24 <f0></f0>	
25F@: 86	46	5C	22	86	46	1F	25 <e8></e8>	l
25F8:06	46	5C	22	96	46	5C	22(86)	
2600:86	42	2E	48	82	02	41	5B <bd></bd>	ı
2600: 46	70	70	C <sub>6</sub>	ØC.	22	46	5C<7F>	ı
2610:22	286	46	EC	23	86	46	EC(1E)	ı
2618:23	86	46	5C	22	26	46	4C <e7></e7>	П
2620:23	66	46	74	23	86	46	50<81>	П
2628:22	96	46	50	22	86	46	5C<16>	ı
2630:22	06	46	SC	22	86	42	ZE <f6></f6>	ŀ
2638:48	02	02	41	90	46	70	70<63>	
2640: C6	90	22	46	SC	22	96	CE(E1)	
2648:46		47	46			86		
	C9			8C	22		46<6E>	
2650:34	22	06	46	SC	22	96	46<11>	
2658: D4	22	186	96	86	46	SC	22<5A>	
2660:06	46	4C	23	86	46	24	23<71>	
2668:06	46	SC	22	86	42	2E	48<8E>	
2670:02	02	41	C5	46	78	70	C6<2D>	
267B: <b>B</b> C	22	46	SC.	22	26	46	SCKE3>	
2680:22	186	46	SC.	22	86	46	SC(DF)	
2688:22	06	46	C4	23	86	46	C4 <fc></fc>	
2690:23	86	46	SC.	22	96	46	ECK1E>	
2698:23	96	46	14	24	86	46	50<93>	
26A0: 22	86	42	2E	48	02	02	41 <f9></f9>	
26A8:F8	46	70	70	C6	ec.	22	46<3F>	
2690:5C	22	96	46	E7	25	86	46<7A>	
2688:E7	25	96	46	5C	22	96	46 <db></db>	
26C0: 4C	23	96	46	74	23	06	46<8E>	
26CB: 5C	22		46	EC				
	24	86			23	86	46<20>	
2600:14		96	46	5C	22	86	42 <fc></fc>	
26D8: 2E	48	02	02	41	2D	47	70<26>	
26E 0: 70	C6	OC.	22	46	SC.	22	89<(92>	
2658:46	95	27	66	86	86	46	50<12>	
26FØ: 22	86	46	EC	23	86	46	ØF<21>	
26F8: 26	96	46	5C	22	96	46	BE<82>	
2766:25	86	46	BF	25	86	46	20<38>	
2708:22	86	42	2E	48	62	02	41<83>	
2710:62	47	70	78	C6	OC.	22	46 <ce></ce>	
2718:5C	22	186	46	E7	25	86	46 <cf></cf>	
2720: E7	25	66	46	2C	22	86	46 <fd></fd>	
2728:5C	22	186	46	E1	26	66	86<73>	
2730:06	06	96	86	86	46	5C	22 <c3></c3>	
2738:86	42	2E	48	82	82	41	95 <b4></b4>	
		70.00	63	-				
2740:47	70	70	CO	ØC	22	46	CA <b7></b7>	
	70 C5	48		22		46		
2740:47			5C	22	86	46	@C <fb></fb>	
2740:47 2748:47 2750:22	C5 86	48 46	SC AC	22 22	86 86	46 46	BC <fb> 5C<c4></c4></fb>	
2740:47 2748:47 2750:22 2758:22	C5 86 86	48 46 46	SC AC BF	22 22 25	86 86	46 46 46	BC <fb> 5C<c4> BF&lt;6A&gt;</c4></fb>	
2740:47 2748:47 2750:22 2758:22 2760:25	C5 86 86 86	48 46 46 46	SC AC BF SC	22 22 25 22	86 86 86 86	46 46 46 46	8C <fb> 5C<c4> BF&lt;6A&gt; 5C&lt;52&gt;</c4></fb>	
2748: 47 2748: 47 2758: 22 2758: 22 2768: 25 2768: 22	05 86 86 86 86 86	48 46 46 46 46	5C AC BF 5C 5C	22 22 25 22 22	86 86 86 86	46 46 46 46	8C <fb> 5C<c4> BF&lt;6A&gt; 5C&lt;52&gt; 5C<e1></e1></c4></fb>	
2749:47 2748:47 2759:22 2758:22 2769:25 2768:22 2779:22	05 06 06 06 06 86	48 46 46 46 46 42	5C AC BF 5C 5C 2E	22 25 25 22 22 48	86 86 86 86 86	46 46 46 46 46 02	9C <fb> 5C<c4> BF&lt;6A&gt; 5C&lt;52&gt; 5C<e1> 41&lt;2B&gt;</e1></c4></fb>	
2740:47 2748:47 2750:22 2758:22 2760:25 2768:22 2770:22 2778:C4	05 06 06 06 06 86 47	48 46 46 46 46 42 70	5C AC BF 5C 5C 2E 78	22 25 22 22 22 48 C6	86 86 86 86 82 80	46 46 46 46 22 22	8C <fb> 5C<c4> BF&lt;6A&gt; 5C&lt;52&gt; 5C<e1> 41&lt;2B&gt; 46&lt;56&gt;</e1></c4></fb>	
2748: 47 2748: 47 2758: 22 2758: 22 2768: 25 2768: 22 2778: 22 2778: C4 2788: 50	05 06 06 06 06 86 47 22	48 46 46 46 42 70 06	5C AC BF 5C 5C 2E 78 46	22 25 22 22 22 48 C6	86 86 86 86 82 80 22	46 46 46 46 92 22 96	0C <fb> 5C<c4> BF&lt;6A&gt; 5C&lt;52&gt; 5C<e1> 41&lt;2B&gt; 46&lt;56&gt; 46<a5></a5></e1></c4></fb>	
2740:47 2748:47 2750:22 2758:22 2760:25 2768:22 2770:22 2778:C4 2760:5C 2788:34	C5 86 86 86 86 47 22 22	48 46 46 46 42 70 06 06	5C AC BF 5C 5C 2E 78 46 46	22 22 25 22 22 48 C6 8C 5C	86 86 86 86 82 80 22 22	46 46 46 46 92 22 96 96	8C <fb> 5C<ca> BF&lt;6A&gt; 5C&lt;52&gt; 5C<e1> 41&lt;2B&gt; 46&lt;26&gt; 46<a5> 46<a5></a5></a5></e1></ca></fb>	
2748: 47 2748: 47 2758: 22 2758: 22 2768: 25 2768: 22 2778: 22 2778: C4 2788: 54 2798: EC	05 06 06 06 86 47 22 22 23	48 46 46 46 42 70 06 06 06	5C AC BF 5C 5C 2E 78 46 46 46	22 25 22 22 48 C6 8C 5C	86 86 86 86 82 80 22 22 22	46 46 46 46 92 22 96 96 96	8C <fb> 5C<ca> BF&lt;6A&gt; 5C&lt;52&gt; 5C<e1> 41&lt;2B&gt; 46&lt;56&gt; 46<a5> 46<a5> 46&lt;15&gt;</a5></a5></e1></ca></fb>	
2740: 47 2748: 47 2750: 22 2758: 22 2760: 25 2768: 22 2770: 22 2778: C4 2780: 5C 2788: 34 2790: EC	05 06 06 06 06 86 47 22 22 23 22	48 46 46 46 42 70 86 86 86	5C AC BF 5C 5C 2E 78 44 44 44 44	22 22 25 22 22 48 C6 8C 5C 8F 4C	96 96 96 96 92 90 22 22 26 23	46 46 46 46 92 22 96 96 96	8C <fb> 5C<c4> BF&lt;6A&gt; 5C&lt;52&gt; 5C<e1> 41&lt;2B&gt; 46&lt;56&gt; 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5> 46<a5 46 46 46 46 46 46 46 46 46 46 46 46 46</a5 </a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></e1></c4></fb>	
2740: 47 2748: 47 2758: 22 2758: 22 2768: 25 2768: 25 2778: C4 278: 50 2788: 34 2790: 60 2798: 50 2740: 74	05 06 06 06 86 47 22 22 23 22 23	48 46 46 46 42 70 06 06 06 06 06	5C AC BF 5C 5C 78 44 44 44 44 44 44	22 25 22 22 22 48 66 80 50 40 50	96 96 96 96 92 90 22 22 26 23 22	46 46 46 46 92 22 96 96 96 86 86	8C <fb> 5C<c4> BF&lt;6A&gt; 5C&lt;52&gt; 5C<e1> 41&lt;2B&gt; 46&lt;56&gt; 46<a5> 46<a5> 46<a5> 46<i5> 46<i5> 46<i5> 46<i5> 46<i5></i5></i5></i5></i5></i5></a5></a5></a5></e1></c4></fb>	
2740: 47 2748: 47 2750: 22 2758: 22 2760: 25 2768: 22 2770: 22 2778: C4 2780: 5C 2788: 34 2790: EC 2798: 5C 2798: 5C	C5 86 86 86 86 47 22 23 22 23 48	48 46 46 46 42 70 06 06 06 06 06 06	5C AC BF 5C 5C 2F 44 44 44 44 44 44 44 44 44 44 44 44 44	22 25 22 22 48 C6 9C 5C 4C 5C 4C 5C 41	96 96 96 96 92 90 22 22 26 23 22 79	46 46 46 46 92 22 96 96 96 86 47	8C <fb> 5C<c4> BF&lt;6A&gt; 5C&lt;52&gt; 5C<e1> 41&lt;2B&gt; 46&lt;56&gt; 46&lt;56&gt; 46&lt;55&gt; 46&lt;15&gt; 46&lt;15&gt; 46&lt;173&gt; 46&lt;173&gt; 46&lt;173&gt;</e1></c4></fb>	
2740: 47 2748: 47 2750: 22 2758: 22 2760: 25 2768: 22 2770: 22 2778: C4 2780: 5C 2788: 34 2790: EC 2798: 5C 2748: 74 2748: 25 2780: 05	05 06 06 06 06 06 06 47 22 22 23 22 23 48 05	48 46 46 46 42 70 06 06 06 06 06 06 05	5C AC BF 5C 5C 2F 46 46 46 46 46 46 92 95	22 25 22 22 48 C6 8C 5C 4C 5C 41 65	86 86 86 86 82 80 22 22 26 23 22 25 85	46 46 46 46 92 22 96 96 96 96 86 47 95	8C <fb> 5C<c4> BF&lt;6A&gt; 5C&lt;52&gt; 5C<e1> 41&lt;2B&gt; 46&lt;56&gt; 46<a5> 46<a5> 46<a5> 46<a5> 46<a73> 46<a73> 42<a7> 91<a73> 95<a75></a75></a73></a7></a73></a73></a5></a5></a5></a5></e1></c4></fb>	
2740: 47 2748: 47 2750: 22 2758: 22 2768: 22 2770: 22 2770: 22 2778: C4 2780: 5C 2788: 34 2790: EC 2798: SC 2740: 25 2740: 25 2780: 05 2780: 05	05 06 06 06 06 06 47 22 23 22 23 40 05 05	48 46 46 46 46 42 70 86 86 86 86 87 85 85	5C AC BF 5C 2E 78 44 44 44 45 95 95	22 25 22 22 48 C6 80 50 40 50 41 65 25	86 86 86 86 82 22 22 26 23 22 F9 85	46 46 46 46 92 22 96 96 96 96 86 47 95 86	8C <fb> 5C<c4> BF&lt;6A&gt; 5C&lt;52&gt; 5C<e1> 41&lt;2B&gt; 46&lt;56&gt; 46&lt;56&gt; 46&lt;55&gt; 46&lt;15&gt; 46&lt;15&gt; 46&lt;173&gt; 46&lt;173&gt; 46&lt;173&gt;</e1></c4></fb>	
2740: 47 2748: 47 2750: 22 2758: 22 2760: 25 2768: 22 2770: 22 2778: C4 2780: 5C 2788: 34 2790: EC 2798: 5C 2748: 74 2748: 25 2780: 05	05 06 06 06 06 06 06 47 22 22 23 22 23 48 05	48 46 46 46 42 70 06 06 06 06 06 06 05	5C AC BF 5C 5C 2F 46 46 46 46 46 46 92 95	22 25 22 22 48 C6 8C 5C 4C 5C 41 65	86 86 86 86 82 80 22 22 26 23 22 25 85	46 46 46 46 92 22 96 96 96 96 86 47 95	8C <fb> 5C<c4> BF&lt;6A&gt; 5C&lt;52&gt; 5C<e1> 41&lt;2B&gt; 46&lt;56&gt; 46<a5> 46<a5> 46<a5> 46<a5> 46<a73> 46<a73> 42<a7> 91<a73> 95<a75></a75></a73></a7></a73></a73></a5></a5></a5></a5></e1></c4></fb>	
2740: 47 2748: 47 2750: 22 2758: 22 2768: 22 2770: 22 2770: 22 2778: C4 2780: 5C 2788: 34 2790: EC 2798: SC 2740: 25 2740: 25 2780: 05 2780: 05	05 06 06 06 06 06 47 22 23 22 23 40 05 05	48 46 46 46 46 42 70 86 86 86 86 87 85 85	5C AC BF 5C 2E 78 44 44 44 45 95 95	22 25 22 22 48 C6 80 50 40 50 41 65 25	86 86 86 86 82 22 22 26 23 22 F9 85	46 46 46 46 92 22 96 96 96 96 86 47 95 86	8C <fb> 5C<c4> BF&lt;6A&gt; 5C&lt;52&gt; 5C<e1> 41&lt;2B&gt; 46&lt;56&gt; 46<a5> 46<a5> 46<a5> 46<a73> 46&lt;15&gt; 46&lt;73&gt; 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47<a73> 47</a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a73></a5></a5></a5></e1></c4></fb>	
2740: 47 2748: 47 2750: 22 2758: 22 2768: 22 2770: 22 2778: C4 2780: 5C 2788: 34 2790: EC 2798: 5C 2798: 5C 2798: 5C 2798: 5C 2798: 5C 2798: 95 2780: 95	05 06 06 06 06 06 06 06 07 22 22 23 22 23 40 05 05 05 05 05 05 05 05 05 05 05 05 05	48 46 46 46 42 79 86 86 86 85 85 85	5C AC BF 5C 2E 78 46 46 46 46 46 82 85 85	22 25 22 22 48 C6 80 50 40 50 41 65 85 85	86 86 86 86 87 22 22 26 23 22 25 85 85 85	46 46 46 46 22 26 86 86 86 86 86 86 86 86 86	8C <fb> 5C<c4> BF&lt;6A&gt; 5C&lt;52&gt; 5C<e1> 41&lt;2B&gt; 46&lt;56&gt; 46<a5> 46</a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></e1></c4></fb>	
2740: 47 2748: 47 2750: 22 2758: 22 2768: 22 2776: 22 2778: C4 278: 50 2788: 34 2790: EC 2740: 74 2740: 74 2740: 74 2740: 75 2740: 75 2740: 05 2760: 05 2760: 05 2760: 05	05 06 06 06 06 06 06 07 22 23 22 23 48 05 05 05 05 05 05 05 05 05 05 05 05 05	48 46 46 46 46 42 70 06 06 06 05 05 05 05 05	5C AC BF 5C 5C 2E 78 46 46 46 46 92 50 95 95 95	22 22 25 22 22 48 C6 80 50 41 85 85 85 85 85	86 86 86 86 87 22 22 23 22 25 85 85 85	46 46 46 46 82 22 86 86 86 47 86 86 86 86 86 86	8C <fb> 5C<c4> BF&lt;6A&gt; 5C&lt;52&gt; 5C<e1> 41&lt;2B&gt; 46&lt;56&gt; 46&lt;56&gt; 46&lt;55&gt; 46&lt;573  46&lt;15&gt; 46&lt;173  95&lt;7E&gt; 95&lt;7E&gt; 95&lt;7E&gt; 95&lt;18&gt; 95&lt;18&gt; 95&lt;18&gt; 95&lt;18&gt;</e1></c4></fb>	
2740: 47 2748: 47 2750: 22 2758: 22 2760: 25 2768: 22 2778: C4 2780: 5C 2788: 34 2790: 5C 2748: 74 2748: 25 2748: 05 2760: 05 2760: 05 2760: 05 2760: 05 2760: 05 2760: 05	05 06 06 06 06 06 07 22 22 23 24 05 05 05 05 05 05 05 05 05 05 05 05 05	48 46 46 46 42 70 06 06 06 06 07 05 05 05 05 05	5C AC BF 5C 5C 2E 78 46 46 46 46 82 85 85 85	22 22 25 22 24 46 66 50 41 85 85 85 85 85 85 85	866 866 862 222 23 222 25 85 85 85 18	46 46 46 46 22 86 86 86 86 86 86 86 86 86 86 86 86 86	8C <fb> 5C<c4> BF&lt;6A&gt; 5C&lt;52&gt; 5C<e1> 41&lt;2B&gt; 46&lt;56&gt; 46<a5> 46<a5> 46<a5> 46<a73> 46<y73> 42&lt;47&gt; 91<y73> 95<ye) 95<y<="" 95<ye="" 95<ye)="" td=""><td></td></ye)></y73></y73></a73></a5></a5></a5></e1></c4></fb>	
2740: 47 2748: 47 2750: 22 2750: 25 2760: 25 2770: 22 2770: 22 2778: 54 2780: 55 2780: 55 2790: 65 2790: 65 2780: 05 2760: 05 2760: 05 2760: 05 2760: 05 2700: 05 2700: 05 2700: 05	25 86 86 86 47 22 23 22 23 48 95 95 95 23	48 46 46 46 46 46 86 86 86 85 85 85 85 85 85 85 85 85 85 85 85	SC AC BF SC 278 44 44 44 44 44 44 44 44 44 44 44 44 44	22 25 22 22 48 C6 8C 5C 41 95 95 95 95 10	86 86 86 82 22 23 22 25 85 85 85 18 80	46 46 46 46 46 86 86 86 86 86 86 86 86 86 86 86 86 86	8C <fb> 5C<c4> BF&lt;6A&gt; 5C&lt;52&gt; 5C<e1> 41&lt;2B&gt; 46&lt;56&gt; 46<a5> 46<a5 a5="" a<="" td=""><td></td></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></e1></c4></fb>	
2740: 47 2748: 47 2750: 22 2758: 22 2760: 25 2768: 22 2770: 22 2778: C4 2780: 55 2788: 34 2790: E5 2798: S5 2780: 05 2788: 35 2780: 05 2788: 35 2700: 05 2700: 05 2700: 05 2700: 10 27E0: 10	22 23 48 95 95 95 23 10 10	48 46 46 46 46 46 86 86 86 85 85 85 85 85 85 85 85 85 85 85 85 85	SC AC BF 5C 5C 2E 78 444 444 445 95 95 32 19 38	22 25 22 22 48 C6 8C 5C 41 85 95 95 95 10 10	866 866 866 867 22 22 22 23 22 25 85 85 85 18 80 18	46 46 46 46 46 86 86 86 86 86 86 86 86 86 86 86 86 86	8C <fb> 5C<c4> BF&lt;6A&gt; 5C&lt;52&gt; 5C<e1> 41&lt;2B&gt; 46&lt;56&gt; 46&lt;56&gt; 46&lt;54&gt; 46&lt;573  46&lt;73&gt; 42&lt;47&gt; 91&lt;73&gt; 95&lt;7E&gt; 95&lt;7E&gt; 95&lt;18&gt; 95&lt;18&gt; 96&lt;18&gt; 96&lt;18&gt; 98&lt;1F&gt; 10&lt;82&gt; 98&lt;16&gt; 93&lt;4E&gt;</e1></c4></fb>	
2740: 47 2748: 47 2750: 22 2758: 22 2760: 25 2768: 22 2778: C4 2780: 5C 2788: 34 2790: EC 27A8: 74 27A8: 2E 2780: 05 27B8: 35 27C0: 05 27D0: 05 27D0: 05 27D0: 05 27E0: 15 27F0: 35	22 23 48 65 65 65 23 10 22	48 46 46 46 46 46 86 86 86 85 85 85 85 85 11 11 11 11	SC AC BF 5C 5C 2E 78 444 444 445 825 85 32 19 38 11	22 22 25 22 22 48 C6 86 5C 41 85 85 85 85 85 85 10 10 80	866 866 866 862 22 22 22 23 22 25 85 85 85 16 85 16 80 18 80 18	46 46 46 46 66 66 66 66 86 47 65 66 86 87 86 86 86 86 86 86 86 86 86 86 86 86 86	8C <fb> 5C<c4> BF&lt;6A&gt; 5C&lt;52&gt; 5C<e1> 41&lt;2B&gt; 46&lt;56&gt; 46&lt;56&gt; 46&lt;54&gt; 46&lt;573  46&lt;73&gt; 42&lt;47&gt; 91&lt;73&gt; 95&lt;7E&gt; 95&lt;18&gt; 95&lt;18&gt; 95&lt;18&gt; 95&lt;18&gt; 95&lt;18&gt; 96&lt;18&gt; 96&lt;18&gt; 98&lt;1F&gt; 10&lt;82&gt; 98&lt;1F&gt; 10&lt;82</e1></c4></fb>	
2740: 47 2748: 47 2758: 22 2758: 22 2768: 22 2778: 24 2788: 56 2788: 34 2798: 56 2798: 56 2798: 35 2708: 35 2708: 35 2708: 35 2708: 35 2708: 35 2708: 35 2708: 35 2708: 35 2708: 35 2708: 35 2708: 35 2708: 35 2708: 35 2708: 35 2708: 35	25 86 86 86 47 22 23 48 85 85 85 19 23 19 22 36	48 46 46 46 46 46 86 86 86 85 85 85 27 10 11 11 12 12 13	SC AC BF 5C 5C 25E 78 446 446 446 446 455 655 655 32 18 38 11 33	22 22 25 22 22 48 C6 8C 5C 4C 5C 41 85 85 85 85 85 85 85 10 10 10 10 10 10 10 10 10 10 10 10 10	866 866 866 867 867 222 224 23 222 23 25 85 85 85 16 86 16 86 16 86 16 86 16 86 16 86 16 86 16 86 16 86 16 86 16 86 16 86 16 86 86 86 86 86 86 86 86 86 86 86 86 86	46 46 46 46 86 86 86 86 87 86 86 86 86 86 86 86 86 86 86 86 86 86	8C <fb> 5C<c4> BF&lt;6A&gt; 5C&lt;52&gt; 5C<e1> 41&lt;2B&gt; 46&lt;56&gt; 46<a5> 46<a5> 46<a5> 46<a73> 42<a7> 91&lt;73&gt; 95&lt;7E&gt; 95&lt;7E&gt; 95&lt;10&gt; 95&lt;10  9</a7></a73></a5></a5></a5></e1></c4></fb>	
2740: 47 2748: 47 2758: 22 2758: 22 2768: 22 2778: 22 2778: 52 2778: 52 2788: 34 2790: 50 2788: 35 2798: 05 2708: 05 2708: 05 2708: 05 2708: 33 2760: 10 2768: 15 2769: 32 2769: 32 2769: 32 2769: 32 2769: 32 2769: 32	05 86 86 86 47 22 23 48 05 05 05 10 22 36 05	48 46 46 46 46 46 46 86 86 86 87 87 87 87 87 87 87 87 87 87 87 87 87	SC AC BF 5C 5C 25E 78 446 446 446 446 446 446 446 446 446 44	22 22 25 22 22 48 C6 5C 41 85 85 85 85 85 85 10 10 10 10 85	866 866 866 867 867 222 224 233 224 235 855 855 865 865 865 866 866 866 866 86	46 46 46 46 46 86 86 86 86 87 86 86 86 86 86 86 86 86 86 86 86 86 86	8C <fb> 5C<c4> BF&lt;6A&gt; 5C&lt;52&gt; 5C<e1> 41&lt;2B&gt; 46&lt;56&gt; 46<a5> 46<a5 a5="" a<="" td=""><td></td></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></a5></e1></c4></fb>	
2740: 47 2748: 47 2758: 22 2758: 22 2768: 22 2778: 24 2788: 56 2788: 34 2798: 56 2798: 56 2798: 35 2708: 35 2708: 35 2708: 35 2708: 35 2708: 35 2708: 35 2708: 35 2708: 35 2708: 35 2708: 35 2708: 35 2708: 35 2708: 35 2708: 35 2708: 35	25 86 86 86 47 22 23 48 85 85 85 19 23 19 22 36	48 46 46 46 46 46 86 86 86 85 85 85 27 10 11 11 12 12 13	SC AC BF 5C 5C 25E 78 446 446 446 446 455 655 655 32 18 38 11 33	22 22 25 22 22 48 C6 8C 5C 4C 5C 41 85 85 85 85 85 85 85 10 10 10 10 10 10 10 10 10 10 10 10 10	866 866 866 867 867 222 224 23 222 23 25 85 85 85 16 86 16 86 16 86 16 86 16 86 16 86 16 86 16 86 16 86 16 86 16 86 16 86 16 86 86 86 86 86 86 86 86 86 86 86 86 86	46 46 46 46 66 66 66 86 47 65 66 86 47 66 86 86 86 86 86 86 86 86 86 86 86 86	8C <fb> 5C<c4> BF&lt;6A&gt; 5C&lt;52&gt; 5C<e1> 41&lt;2B&gt; 46&lt;56&gt; 46<a5> 46<a5> 46<a5> 46<a73> 42<a7> 91&lt;73&gt; 95&lt;7E&gt; 95&lt;7E&gt; 95&lt;10&gt; 95&lt;10  9</a7></a73></a5></a5></a5></e1></c4></fb>	

```
2818:05 05 05 05 05 07 05 05(70)
2820:05 05 05 05 05 05 03
                            79<D7>
2828:70 C6 9C
              22
                  46
                     SC
                         22
                            86<2F
2830:46
        ac.
           22
               86
                  46
                      84
                         22
                            B6<D9>
                  46
2838:46
        5C 22 06
                     AF
                            BAKSE'S
                         25
2848:86 86 46
              50
                  22 86 46
                            C6<RE>
                        86
2848:48 71 49
               31 27
                            86<58>
                     86
2858:46 5C
           22
                  42
                     2E
                         48
                            82<CC:
              86
2858:02
        41
               48
                  70
                     70
                         C6
                            BCKFD2
                            22<74>
2868:22 46 5C 22 06
                      46
                        PC:
2868: 86
        46
           34
               22 86
                     46
                        50
                            22(09)
               25
                            25<67>
2878:86 46 BF
                  26
                     46 BF
              23 86
2878:86 46
           9C
                     46
                        31
                            27<D1>
2880:06
        86 86
               46
                  50
                     22
                         86
                            42(32)
2888: 2E 48 82 82 41
                     D7 48
                            79<8A>
                     50
                            86<97>
5C<4D>
2890:70 C6 0C
               22 46
                        22
2898:46 31 27 86 86 86 46
                            74(99)
28A0:22
        96
                  23 86 46
           46
               4C
28A8: 23 86 46
               9C
                  23 06
                         46
                            24<FF>
                  23 96
2880:23 96 46
               74
                         46
                            5C(2C)
2888:22 86
           42
               2E
                  48 82
                        92
                            41<4D>
2800: 8A
        49
           78 78 C6
                     OC.
                        22
                            46<65>
28C8:5C 22 86
               46 4C
                     23
                         26
                            46(24)
        23 86
               46
                  9C
                     23
                            46<@B>
28D8: 9C 23 86 46
                     25 B6
                            46(33)
                  1F
28E0:50
        22 26 46
                  90
                     23 86
                            46<073
                  90
        23 86
               46
                     23 B6
                            42<9A>
28E8:9C
28F0: ZE
        48 02
              B2 41
                     30
                         49
                            72(09)
28F8: 49 6D 4A
              A2 FF
                        20
                            85<9D>
                     9A
2908: 2C 20 96 48 4C C3 2A
                            29(99)
        2D A9
2998: 44
               48 BD
                         04
                            A2<9E>
                     GUE
2918:FF
        9A AD
              59
                  2A 80
                         72
                            4B(37)
                            88KEA>
2918:38 E9 11
              8D 75
                     38 A2
2929:49
        10 90
              50 48 EB EB
                            RITCEE >
2928: DØ F6 A9
               13 BD
                     7C
                         48
                            A2<AB>
2930:00
        8E 1D D0 8E
                     2F
                         22
                            8A< 7D)
29KR: 90 DD 44
              E8 EØ BC DØ
                            FR(25)
2948: A9
        10 80 AR 48 80 AC
                            49(59)
2948: A9
        78 SD
              07 D4
                            80<43>
                     89
                        83
2950:0A
        90
           49
               18 BD
                     ØC
                         DØ
                            A9<9B>
2958:01 80 6F 62 A9
                     94
                            C3<D8>
                        BD
2960:02
        A9 16 80 CR 92
                        A9
                            1C(D4)
2968:80 C1 82 A9 86 80 C2 82<38>
                        A4
2970: A9
        78
           BD
              F4 82 A9
                            SD: SD:
2978:C6
        82 AE
               75
                  26
                     BD D6
                            2F<8E3
2980:8D C4 02 BD DD
                     2F
74
                        80
                            C3<95>
2988:02 BD E4
              2F
                  BD
                        340
                            A9(BE)
2990:16
        BD
           C5
              02 A9 FA BD
                            C7<DE>
2998: 82
        29
           93
               3C 2Ø
                     15
                        3D
                            20<20>
29A8: FD
        38
           20
              26 36
                            38(49)
                     AE
29AB: BD C7
           2F
              SD 71
                     39 BD
                            CE(AR)
2980:2F
        BD
           79
              38 8D
                     88
                            BD<80>
                        DØ
2988:01
        00
           26
               86
                  20
                     66
                         37
                            4C<C62
2900:30
        40
           20 E8 36 A9
                        68
                            8DKF6>
29C8: 7E
        38 A9
              80 SD 1F D8
                            RINCACO
2900: 69
        20
           BD
              86
                  39 BD 87
                            30KEE>
29D8:8D
        7F
           30
              8D 81 38 8D
                            CB(13)
29FB: 82
        SD
           7A
              30 85
                     14
                        A9
                            03<59>
29E8:80 ID D0
              AS CO BD MA
                            PRZ(EA)
29F 8: A9
        28 60
              191 192 A9 CB
                            6E(06)
29F8: 4A
        DC
           40
                     D4
              BD BE
                        A5
                            14(97)
2A00:C9 02 D0 FA A9
                            14<3B>
                     91
                        85
2008:28 13
           339
              20 B1
                     339.
                        20
                            SSCDC)
2A18:38 28
           SE
               37
                  20 DB
                        33
                            20(28)
                  30
2A18:F2 37
           20
              59
                     28
                        Œ
                            35(FF)
2A20: 20 94 35
              AD
                  ZA.
                     314
                         C9
                            14<D4>
2A28:90 RE A9
              14 BD
                     7A 30
                            20<09>
2A30:12 35 20
              Æ
                  35
                     28
                        34
                            35<76>
2A38: EE
        78
           30
              20 D6
                     34
                            59(42)
                        20
2A48:3E AD 87
              30 C9
                     98
                            09(91)
2A48: 20 D3
           30
              26
                  5E
                     3A
                        20
                            3A<80>
2A50:3D 20 E6
              35 29 48
                        330
                            ADK 823
2A58:87
        30 C9
               88 FB
                            95(02)
                     86
                        29
2A68: 3D
        28
           98
                  20
                     4B
                            4C<6E>
2A68:71 4A 80 78 FB 78 80
                            00<05>
2A78:00 00 00
              80 28
                     98
                        PIPI
                            PROCEAS
        1F
           15
2676:00
               18 18 18 18
                            18(63)
2A80:18 1F 1F
               88
                  88
                     86
                        18
                            18(D4)
2A88:18 F8 F8
              88
                  88
                            98KE5>
                     90
2A99: F8 F8
              19 18 11
                        346
                            BB(84)
2A98:00 FF
           FF
              88
                  000
                     66
                        200
                            88KBC>
2AA6: 00 FF
               18
                  18
                     18
                         18
                            18(7F)
2AA8: 18 FF
           FF
              88 88
                     88
                        18
                            18<76>
2ABB: 18 18 18
              18 18
                     18
                        753
                            18(02)
2AB9: 18
        1C 1C
              iC
                  7C
                     20
                        (36)
                            66(86)
2AC0:30 FF
           30
              66
                  99
                     22
                            18<F7>
2AC8: 18 7E
           18 18 88 88
                        22
                            99<9C>
2ADM: NA AM AM
              18 18
                     30
                            ののくつつつ
                        PIÓ
```

2AEB:00 00 00 18 18 00 03 04<82> 28E8:80 18 38 68 49 89 7F A3(A8) 2AF0:63 63 63 63 /F 38 18(A5) 2AF 8: 18 18 3F 3E 3E @3<DB> 2800:03 7F AR AR **7F** Æ, 86<A6> 7F 70 2B0B: 06 87 27 **7**F 00 70<9F> 2B10:78 7F 60(86) 78 77 7F 07 22 7F 2B18:68 7C 6CKE72 7F 88 2B28: 68 75 63 63 83<66> 2828:03 1F 18 18 18 00 Œ 36<7F> 2B3Ø: 36 7F 77 77 **7F**-(2)(2) **7**F A3KER> Æ 87 87 30 2B38: 63 87 88 30<50> 2848:3C 00 3C 3C 3C 88 3C 30(80) 2848:3C 20 30 30 18 310 06 **BC**<793 2858:18 38 18 8C 86 88 22 7E<18> 2858: 00 00 7E 80 00 1913 30<6F> 66 28AR-18 RC 18 38 AR RO 7E ASCIRN 2868:83 1F FC 78 DF 79 10 BB<22> 2878:10 88 18 18 18 18 18 18(C2) 33 33 2878:18 18 3F **7F** 73 73<EB> Æ 66 2888:73 88 7F 67 67<93> 2888:7F 88 7F 67 67 68 63 63(36) 2998: 7F 00 TE 66 66 77 17 77(CD> 2899: 7F 20 Æ 68 68 **7**F 718 70(61) 2BA0:7F 00 7F 60 60 **7F** 70 78<59> 6F 29A8:70 00 7F 63 60 67 67<AD> 7Ë 2889:7F 20 73 73 73 73 73<5A> 2BBB: 73 00 **7**F 10 1C 10 10 1C(1F) 28C0: 7F 80 8C 8C 6C 86 SEK 3AX 2BC8: 7E 00 66 66 6C 7F 67 67<F1> 28D0: A7 00 30 33. 300 78 79 78<66> 77 2808:7E 00 7F 67 **7**F 67 67<4D> 28E0: 67 80 67 77 7F 6F 67 67(33) 63 63 2BEB: 67 00 7F 67 67 67<DA> ZBER: JE BR 75 EA EA 7E 70 78(11) 7F 2BFB: 79 88 63 63 67 67(16) 67 2000: 7F 87 7E 66 66 7F 77<FA> 2028:77 00 Æ 68 7F 73 73<01> 83 2C10:7F 86 Æ 1C 1C 10 10 10KPF> 2C18:1C 88 67 67 67 67 67 67(C0) 2C20:7F 292 67 67 67 67 **6F** 3E (EE) 2C28:1C 00 67 **7**F 67 67 6F 7F<58> 2030:67 78 73 73 73 3E 67 67<7C> 2038:67 90 67 67 67 7F 10 10(29) 2C40:1C 00 **7F** 66 6C 18 67<8E> 37 2C48:7F 80 1E 18 18 18 18 18<0E> 2C50: 1E 00 00 80 FB 89 90 00(95) 2058:00 00 00 00 00 00 FF FF(AD) 2C68: FF EE ΕĒ FE FF FE FE(B9) 88 2C68: FE FE FE FE FE F7(B3) 2C70:F7 F7 F7 F7 F7 88 FF FF(85) 2078:66 66 66 66 66 66 66 AACIRS 2080:66 66 66 66 66 66 66 66(53) 2088: 66 66 66 FF FF 99 00<7B> 66 60<DD> 2070:00 00 00 00 0F 1F 22 2098: NO NO NO NO CO CO DE @B< 1D> 2CA0:1F 03 0F 02 02 182 CB CØ<22> 2CA8: CØ CØ CB E0 EØ 1F 20(B0) 20 2080:4F 3D 0F 1D 3F 7D CØ ABKA2> 2CB8:98 ER 80 CB ER F0 88 00<0B) 2008:88 88 88 88 C8 C8 255 Ø5(2C) 10 2008:05 05 05 05 10 85 05<41> 2CD0:05 1D 1D 01 01 01 82 07<BC 2CD8:FD FF 45 11 FF 88 20 00<20> 2CE0:55 FF FF FF 42 C6 88 88<9F> 2CEB: 48 FB FE FF 88 202 C2 77<08> 2CF0:3D 1F 07 01 FF 66 22 2CF8:55 FF FF EE 42 63 20 00(2A) ZD00:15 BF F8 E8 89 88 29(2) 20<91> 2008:0A 1F 7F 20 20 88 08<0F> FF 2D10:54 FF 1F 87 82 88 48 EØ<88) 20 2D18: BE FF A2 88 FF 47 EE < 400 > 2020: BC F0 B0 BA FF 90 90 DMCF45 99 2029:18 66 3C 5A 9B 24 42(FE) 2030:81 99 66 3C 14 24 **00**<63> 2D.39 - 00 18 7E FF Æ 18 18 7E<3A> 2D40: FF 7E 18 88 88 88 (20) PR(RC) 2048:86 FF 1E 18 88 24 42 88< D8> 2D50:00 96 FC 80 47 81 20 00(62) 2058:88 88 88 88 00 ONCE (A) 88KB2> 2DAB: 88 712 3C 18 42 7F FE FECACO 3C 2D68: 7E 42 24 81 30 3C 3C(C8) 2070:41 A4 76 C3 DF C3 FB FB(32) 2078: C3 76 28 18 3R 44 44 44(2E) 2D68: 44 38 19 92 92 FE 10 10(9A) 2088:10 7C FF FF 90 20<93> FF FF 2090:FF FF 66 24 28 18 7E DB<6F> 2D98: A5 66 24 99 C3 66 18 18 (D8) 2DA0:24 55 E0 02 E1 02 00 22(73) Laenge 11688 Bytes

Listing zu »Submission» (Schluß)

88

88<88>

2ADB: 00 7E 00 00



### Vorsicht Falle!

Sie drehen in einem Autoscooter Ihre Runden. Plötzlich rast Ihnen ein anderer Scooter entgegen. Aber keine Panik, »Crazy Scooter« ist nur ein Spiel.

eder fährt gerne einmal mit einem »Bleifuß« auf dem Gaspedal. Aber bitte nicht auf echten Straßen. Wenn Sie sich so richtig austoben möchten, dann starten Sie doch Ihren Computer und laden Sie das Spiel »Crazy Scooter«. Hier können Sie Ihre Reaktion unter Beweis stellen.

Crazy Scooter« ist zu 100 Prozent in Maschinensprache geschrieben. Es stellt eine Scooterbahn auf dem Rummelplatz dar. Sie steuem das blaue Fahrzeug, der Computer das rosarote. Ihre Aufgabe besteht darin, alle Punkte auf der Fahrbahn einzusammeln. Achten Sie aber auf das computergesteuerte Auto! Bauen Sie einen Crash, ist eines Ihrer drei Leben verloren. Rechts neben der Scooterbahn befindet sich Ihre Spritanzeige. Geht Ihnen der Sprit aus, ist's ebenfalls für ein Auto vorbei.

Sollten Sie dem anderen Auto einmal blitzschnell ausweichen müssen, betätigen Sie einfach den Feuerknopf. Ihr Auto beschleunigt dann augenblicklich, was sich allerdings auf den Spritverbrauch auswirkt. Achten Sie also auf die Benzinzufuhr.

Haben Sie trotz Schwierigkeiten alle Punkte eingesammelt, gelangen Sie in die nächsthöhere Spielstufe. Ab dem dritten Level kommt eine neue Schwierigkeit ins Spiel. In der Spielfeldmitte erscheint eine Zahl, die die Anzahl der einzusammelnden Bonuspunkte angibt. Diese Punkte erscheinen aber ganz willkürfich an irgendeiner Stelle der Bahn. Für jeden eingesammelten Bonuspunkt bekommen Sie 50 Punkte. Sollten Sie eine Spielstufe mit zu wenig Bonuspunkten beenden, verlieren Sie wiederum einen Scooter. Ab Spielstufe sechs wird es noch interessanter. Dann behindern Sie noch Straßensperren, denen Sie geschickt ausweichen müssen.

Jeder Level unterscheidet sich vom vorhergehenden. Immer wieder gibt es neue Hindernisse. Stellen Sie doch eine eigene Highscore-Liste auf, denn nur wenige werden die Stufe zehn erreichen.

(Oliver Schmitt/wb)

PROG	RAMM-STECKBRIEF
Programmname	Crazy-Scooter
Programmtyp	Speel
Programmiersprache	Maschinensprache
Programmlänge	6030 Byte
für Computer	800 XL/130 XE
zusätzliche Hardware	Diskettenlaufwerk oder Kassettenrecorder und Joystick
Eingabehilfe	AMPEL
Bemerkung	Sehr schnelles Aktion-Spiel
Leserservice	Diskette (SCOOTER.COM)

0000:FF FF 40 90 38 91 20 08<04>
0008:90 20 EC 91 20 70 99 20<AB>
0010:1E 92 20 38 92 20 E9 9C<BA> 0018:A9 EB BD 30 02 **A7** 88 8D(AF) F4 0020:31 02 A9 88 8D 02 ABC1E> 0028:50 A2 26 A9 20 5C E4<99> 07 0030:A9 FA 8D 1A 82 A9 ÇØ BD<15> BUISH: DE DA 20 2E 92 28 RØ. 93(20) 92(39) 28 BD 01 D2 4C D1 0040:A9 0048: AD 84 DØ 02 C9 00 ØE. 0050: 2B 90 SD 8A 90 49 30 SDC 445 A9 8058:00 D2 4C 9C 92 5A BD(BC> 0060:00 DZ 4C 9C 92 AB 02 BIKE4) 4C 88 90 26 45 99 A0(D8) 99 0070:00 BI CB AA 20 53 40(50) 0078:D8 90 28 Bi CB 09 03<28> AB C9 0090:F0 4C C9 64 FØ 4B 09<602 00881F6 29 09 00901F0 39 60 2F 20 FØ 6F 28 99 C9 38<00> AC DO(DE) AA 20 0098: B1 CB 61 EØ Ø3KAB> 09<E6> 00A0:F0 2C EØ Ø4 FØ 28 60 2F 4C FØ 00ABLF0 09 693 08 ED 38(79) CØ AP 95 00<63> 9989:F9 19 68 0088:8D 32 90 20 BØ 95 EE 32(01) 08 68 0000:90 AD 32 98 C9 05 F3<200 88CB: 4C 89 92 4C 98<82> 7F 91 4C 47 91 AD 05 84:0000 68 00D8: C9 08 FØ **OB** C9 97 FØ OCKFC> 4C A9 RAFA: C9 RF FR RD 30 91 69(7E) 4C 41(40) 41 91 00E8:02 4C @1 A9 84 4C 91 A9 DOFD: 91 41 91 00FB:4C 41 68 48 AD 99 98(17) 0100:C9 01 30 91 37 92 FR 894905 90 0108:4C C8 92 CD 84 FØ EF(72) 0110:60 AD 0A D2 29 C9 03 00<F2> 0118:F0 08 09 @1. FØ 80 Ç9 02<41> 0120:F0 00 4C 6A 91 09 BR 4CKEA> 0128:6C 91 A9 87 AC 6C 91 49<0A> 9130:0E 4C 6C 91 A9 ØD BD 0138:98 A9 01 8D 09 90 20 0F(20) 0140:91 A9 00 8D 09 92 4C D1<BD> 90 8D 02<C0> 0148:92 A9 20 OD 24 0150:D2 20 9E 28 07 94 20<45> 29 94 29 74 95 20 EE 9E<06> 0158:F1 0160:20 39 94 4C 82 9C AD REC4B> 0168:90 C9 9D F0 01 60 89 88KEF) 0170:80 01 D2 80 03 D2 AD 2C(F4)

DØ 09 20 EE 4C 02 93 A9 8178:90 C9 00 D0 09 9E(F4> 9E 01B0:20 58 90KE91> 90 20 9E 71(75) Ø188:8D 24 20 20 EE 28 39 94 A9<44> 0178:94 28 9E 0198:01 1900 20 98 20 31 910 A9< 85> 20 4C 01A6:00 BD 90 02 93 A9<9E> A9 8A AA OB 81AB: 18 BD BA 0180:AB SIG FID oc: 86 AB 49 PMC325 A9 86 BD 81 @188:8D @@ 82 82<46> 01C0:A9 810 28 02 49 9E BDKE63 @1CB: 29 82 49 81 8D 84 D4 BD(AE) 9100:23 98 60 AD 2C 99 69 00K5C> 2C 98 AD 01D8:D0 01 60 CE 20(14) 69 10 19 80 20 BC 98 20 99 32 20 99 01EB: 20 4A 96 B9(12) 01F0:99 28 F4 68(92) @1F8: A9 BD D0(C5) 33 10 0200:C9 06 38 92 93 00 F4<16> 8208:68 A9 98 8D 1D(92) 88 8D 0210:90 16 90 0218:01 OD 1A 90 49 DA BD 1B(CD> 14 BD 77 AD< 78> 0220:90 68 A9 70 98 **C**7 84 FB 24 6B(84) 0228: 0A 26 0230:92 60 20 A5 92 AD EE ZØKAD> 0238:90 AE 20 98. 69 202 9D 88K3D> 0240:83 E0 B4 10<42> FØ EB 88 AR 45 4C 94 BD(86) 0248:01 60 A9 0250:24 98 80 802 D2 '20 29 9E < 099 > 74< 77> 0258:20 20 94 20 F1 94 28 9E 0260:95 20 39 94 4C(RS) 20 EE 90 22 9C CE 22 AD 90<57> 0270:F0 B8 60 EE 20 98 AE 20KE3> 8278:98 A9 88 90 80 80 E0 B4<FA> 90 0280:F0 CB EE 20 AE 28 98<72> 0288:A9 90 9D 99 83 E8 84 FØ<882> 0290:BC E0 AB 10 B5 60 AD 78<F8> 8298:82 8D 85 98 28 QF. 91 20KEB> 90 88 85 4D AD @5<B4> 92A9: 82 A9 02A8: 90 **C9 GB** FO OB. C9 FB<Ø1> 02B0:0D C9 0E F8 PIE 4C FC 92<61> D1 92 0288:20 CD 20 81(25) 4C D1 92 72 20 4C(37) 0200:96 92 97 4C D1 92<54> 02C8: D1 20 1A 0200: AD 1A 90 C9 89 FØ. 3A EE (9A) 7C 90 0208:1A 90 28 90 20 1E 92(D5) 02E0:18 AD 1A 10 D5<D1> 69 8D

02F0:1A 90 C9 03 F8 38 84(11) 0.9 85 FB 02F8:F0 3F 43 C9 BA<32> 34 93 2F 94 **C9** 0300: FB 07<61> 8388:F0 5A C9 Fe C9 89<C9> 98 SE 77 1B 9319:F9 62 40 99 A9 82 ADKSB> 90 810 2318:28 90 A9 BD(B1> 11 0320:05 8A 10 8D D6 89 SD(47) B328:70 90 20 12 92 49 PES 93 A9 2330:2C 4C A7 98 21 BD(2E) 40 **A7** 93 A9 0338:2C 90 82 BD<28> 8340:2C 90 4C **A7** 93 A9 92 8D<80> 49 03 BD 2C 0348: 2B 90 90 4C<AE> 93 A9 ØA BD 18 90 49<9E> 8358: A7 0358:04 8b 28 90 **A7** A7 93 A9 02 93 A9 03 0360:90 4C an 20<03> 4C 2CKBF> 0360:90 8D 4C A7 93 0370:90 A7 94 gp 2C(BB) 0378:90 18 69 10 BD DD 4C<B9> 89 90 A9 AA 0380:77 80 01 D2 A2<E4> 93 20 0388:00 BD 02 BD D2 A9(1A) 0390:00 65 14 A9 FF BD DØ<3F> C9 02 00 FS 0398:05 14 ER ERCR35 60 1D 1F 28(3A) 03A0:16 00 Eá 23 03A8:2D 2F 35 2C 40 48 51 79 A2 86 03B0:60 6C 88 98 A2 C1(32) 8388: D9 F3 20 BD F6 9D(BC) 93 Ø3CØ: D5 69 EB EØ 11 00 FS 60<70> Ø308:27 ØØ 21 9 20 20 25 Ø4<3C> 36 **60** 15 94 9309:99 94 2E 90 25 200 (B9) 03D8:32 A2 66 9D DEK4C> BD ØB DØ 03E0:8A E8 EØ FS 60 03E0:23 28 03F0:30 26 37 21 23 28 2B 2F (42) A2 00 BD 2E 94 PD(AF) 8B D8 03FB: DC BA EB EØ. FB 60(28) 94 219 95 34 00<52> 0400:2F 35 30 26 DC 0408:2F 26 88 35 25 20 A9(D1> 8410:00 AA 9D BA EB EØ OB<AE> 0418:D0 F8 60 AP EF

Listing zu »Crazy Scooter«. Bitte mit AMPEL eingeben.

90 CE 1B

02E8:8A CE 18

0420: AD	25	90	C9	01	FØ	18	A9<37>
0428:BA	80	Ci	02	A9	01	8D	25<38>
8430:98	A9	98	8D	02	02	60	A9 <f4></f4>
0438:38	BD	CI	82	A9	88	80	25<7B>
0440:90	49	45	80	02	02	60	A2<95>
0448:00	BD	7F	94	90	DD	BA	E8<24>
0450:E0	29	DØ	F5	66	2Ē	2F	66 <ed></ed>
0458:00	22	2F	2E	35	33	20	12<66>
0460:95	AD	26	90	C9	88	FØ	01(82)
8468:60	20	EØ	94	AD	ØA.	02	29(C7)
0470:01	C9	01	FØ		20	D9	94 <f1></f1>
	35	89	C9	10	FØ	01	48 <bc></bc>
0478:8D			35				90(84)
0480: A9	2F	9D		89	BE	27	
		SD	26	90	80	28	90(50)
0490:60	20	09	94	BD	75	BA	C9KBA>
0478:00	F@	01	60	A9	2F	9D	75(68)
04A0: BA	BE	27	90	A9	01	8D	26(09)
04A8: 90	A9	02	8D	28	90	60	AD<2B>
0488:0A	D2	29	ØF	AA	60	CE	2AKE4>
0498190	AD	2A	90	DØ	86	A9	64<7E>
04C0:80	24	90	90	68	98	68	AD <d4></d4>
<b>04C8:26</b>	90	C9	21	FØ	21	60	AD(E2>
04D0: 2B	98	C9	02	FØ	29	A9	00 <fe></fe>
0408: AE	27	90	9D	35	89	60	A9(62)
84E8:88	AE	27	98	9D	75	BA	68<64>
04E8: AD	1A	90	C9	62	10	02	68<3@>
04F0:68	60	20	95	95	AD	2F	98<1F>
04FB: C9	20	FB	01	60	20	9F	95<4A>
05001AD	88	20	95	27	96	D2	29<8E>
0508:01	C9	21	FØ	10	20	60	95 <b3></b3>
0510:BD	5F	89	C9	22	FØ	01	60KSF>
0518:A2	38	9D	5F	89	8E	30	90(09)
0520:A9	01	80	2F	90	BD	31	99(68)
0528:60	20	6D	95	BD	4F	BA	C9<66>
8220:00	FØ	01	60	A9	38	90	4F<@C>
2538:8A	SE	20	98	A9	01	aB	2F(53)
				31			ADKB9>
0540:70	A9	22	BD		92	68	
0548: 0A	D2	29	OB	AA	60	AD	2F <fe></fe>
0550:98	CA	21	FØ	21	60	AD	31<83>
2558:92	C9	02	FØ	89	A9	90	AE <34>
0560:30	90	9D	5F	89	60	A9	80 <e6></e6>
2568: AE	20	90	90	4F	8A	60	AD<1D>
0570:1A	90	C9	06	10	02	68	68<65>
0578:60	CE	2E	90	AD	2E	92	D0 <f2></f2>
0580:06	A9	64	BD	ZE	90	68	68<15>
Ø508:6B	60	9	98	BD	26	90	20<30>
0590:74	95	A9	88	8D	2F	90	4C <d9></d9>
0598:C3	95	EE	ØF	90	EE	ØB	90 <fa></fa>
05A0: AD	ØB	90	C9	ØA	DØ	ØD	A9 <b2></b2>
05A8:00	BD	<b>ØB</b>	90	A9	16	BD	ACKE2>
2580:8A	20	E.4	95	18	AD	ØB	90<87>
0598: 69	10	80	AC	8A	60	EE	@C<7D>
05C0:90	AD	ØC	90	<b>C9</b>	ØA.	DØ	@D<98>
Ø508: A7	440	80	<b>SIC</b>	70	47	10	8D(6D)
0500: AB	BA	20	05	96	19	AD	BC<59>
0508:90	69	10	8D	AB	BA	60	EE <cd></cd>
05E0:00	92	AD	ØD	90			
Ø5E8: ØD					C9	MA	DØ<43>
	A9	20	BD	<b>8D</b>	90		10<88>
05F0:80				QD	90	Α9	10(88)
	AA	BA	8D 4C		90 96	A9 1B	10<88> AD<56>
05F8: 0D	AA 90	8A	8D 4C 10	9D 26 8D	90 96 AA	A9 18 8A	10<88> AD<56> 60<61>
05F8: 0D 0600: EE	90 0E	8A 69 28	8D 4C 10 96	00 26 80 23	90 96 AA 97	A9 18 8A 98	10<88> AD<56> 60<61> AD<26>
05F8:00 0400:EE 0408:0E	AA 90 0E 90	8A 69 28 C9	8D 4C 10 96 0A	9D 26 9D 23 90	90 96 AA 97 00	A9 18 8A 98 A9	10<88> AD<56> 60<61> AD<26> 90<85>
05F8:00 0600:EE 0608:0E 0610:8D	90 0E 90 0E	8A 49 28 C9 90	8D 4C 10 96 9A A9	90 26 90 23 00 10	96 96 AA 97 90 80	A9 18 8A 98 A9 A9	10<88> AD<56> 60<61> AD<26> 90<05> 8A<91>
05F8: 0D 0600: EE 0608: 0E 0610: 8D 0618: 4C	90 0E 90 0E 47	8A 69 28 C9 90 96	8D 4C 10 96 2A A7 18	9D 26 9D 23 99 10 AD	96 96 AA 97 80 80 8E	A9 18 8A 98 A9 A9 A9	10(88) AD(56) 60(61) AD(26) 90(85) 8A(71) 69(6E)
05F8: 0D 0400: EE 0408: 0E 0410: 8D 0418: 4C 0420: 10	90 0E 90 0E 47 ED	8A 69 28 C9 90 96 A9	8D 4C 10 96 2A A9 18 8A	9D 26 9D 23 D9 10 AD 60	90 96 AA 97 00 80 80 40	A9 18 8A 98 A9 A9 A9	10(88) AD(56) 60(61) AD(26) 90(05) 8A(91) 69(6E) 96(55)
05F8: 0D 0600: EE 0608: 0E 0610: 8D 0618: 4C 0620: 10 0628: A9	90 0E 90 0E 47 6D 88	8A 49 28 C9 90 96 A9 8D	8D 4C 10 96 0A A9 18 8A 00	9D 26 9D 23 D9 10 AD 60 90	90 96 4A 97 80 80 80 40 80	A9 18 8A 98 A9 A9 47 Ø3	10<88> AD<56> 60<61> AD<26> 90<85> 84<91> 69<6E> 96<55> 98<84>
05F8:00 0400:EE 0408:0E 0610:8D 0618:4C 0420:10 0428:49 0430:49	90 0E 90 0E 47 6D 81	8A 69 28 C9 90 96 A9 8D	8D 4C 10 96 2A A9 18 8A 80 19	9D 26 9D 23 D9 10 AD 60 90 96	90 96 AA 97 80 80 80 40 80 A9	A9 1B 8A 90 A9 A9 90 47 Ø3	10<88> AD<56> 60<61> AD<26> 90<85> 8A<91> 69<6E> 96<55> 90<84> 8B <ab></ab>
05F8:00 0400:EE 0408:0E 0610:8D 0418:4C 0420:10 0428:49 0430:49	AA 90 0E 90 0E 47 6D 86 81 90	8A 69 28 C9 96 A9 8D 8D 8D	8D 4C 10 96 2A A9 18 8A 20 19 16	9D 26 8D 23 DØ 10 AD 60 90 90	90 96 97 97 90 80 80 40 80 40 80 40 80	A9 18 8A 98 A9 A9 47 83 82 87	10<89> AD<56> AD<56> 60<61> AD<26> 20<405> 88<91> 69<66> 96<55> 96<84> 80<85>
05F8:0D 0600:EE 0608:0E 0610:8D 0618:40 0620:10 0628:47 0638:04 0640:05	AA 90 0E 90 0E 47 ED 01 90 90 90	8A 69 28 69 96 A9 8D 8D 8D A9	8D 4C 10 96 8A 8A 8A 80 17 16 8A	90 26 80 23 09 10 40 40 90 90 80	90 96 97 90 80 80 80 40 80 40 80 99	A9 18 8A 90 A9 A9 90 47 03 02 87	10<69> AD<56> AD<56> 60<61> AD<26> 80<85> 8A<91> 69<6E> 96<85> 98<83 80<37> AD<37> AD<37 AD<37 AD<37 AD<37 AD<37 AD<37 AD
05F8:0D 0600:EE 0610:8D 0610:8D 0618:4C 0620:10 0628:47 0638:49 0638:49 0640:05 0648:FF	AA 90 0E 90 0E 47 6D 08 01 90 8D	8A 8D 8D 8D 8D 8D 8D 8D 8D 8D 8D 8D 8D 8D	8D 4C 10 96 8A A9 18 8A 8A 19 16 8A 98	90 26 90 23 00 10 40 90 90 90 80 80	90 96 97 90 80 80 80 40 80 80 80 80 80 80 80 80 80 80 80 80 80	A9 18 8A 98 A9 A9 47 83 82 87 98 80	10<89> AD<56> AD<56> 60<61> AD<26> 20<805> 8A<91> 69<6E> 96<55> 90<84> BD <ab> BD<ab> AD&lt;37&gt; AP&lt;05&gt; 2A&lt;80&gt;</ab></ab>
05F8: 0D 0600: EE 0610: ED 0610: ED 0618: 4C 0620: 10 0628: A9 0630: 04 0640: 05 0648: FF 0650: 90	AA 90 0E 90 0E 47 8D 8D 8D 8D	8A 49 28 C9 96 A9 8D 8D A9 13 2E	8D 4C 10 96 9A A9 18 8A 8A 90 19 16 9A 90	90 26 80 23 10 40 40 90 90 80 80 80 80	90 96 97 90 80 80 80 40 80 99 99 99 64 21	A9 18 8A 90 A9 90 47 03 02 87 90 8D 8D	10<89> AD<56> AD<56> 60<61> AD<26> 90<85> BA<91> 69<6E) 96<55> 90<84> BD <ab> BD<ab> AD&lt;26 BD<ab> BD<ab> BD<ab> BD<ab> BD<ab ab="" bd="" bd<="" bd<ab="" td=""></ab></ab></ab></ab></ab></ab></ab>
05F8: 0D 0600: EE 0610: 8D 0610: 8D 0618: 4C 0620: 10 0638: 9 0638: 84 0640: 05 0648: FF 0658: 90	AA 90 0E 90 0E 47 ED 81 90 8D A9	8A 69 78 96 96 80 80 80 49 13 2E 14	8D 4C 10 96 8A 8A 8A 80 19 16 8A 90 90 8D	90 26 90 10 40 90 90 90 80 80 80 80 80 80 80 80 80 80 80 80 80	90 96 97 80 80 80 80 80 80 80 80 80 80 80 80 80	A9 18 8A 90 A9 90 47 93 87 90 80 80 80 80	10<88> AD<56> AD<561> AD<561> AD<26> 00<61> AD<26> 00<85> 8A<91> 49<65> 96<55> 96<84> 8D<85> AB<37> AP<884  AB<37> AP<85> 2A<800  20<0000000000000000000000000000000
05F8: 0D 0600: EE 0610: ED 0610: 8D 0618: 4C 0620: 10 0638: 04 0638: 04 0640: 05 0648: F0 0658: 90 0658: 90	AA 90 0E 90 0E 47 ED 011 90 8D 8D A9 97	8A 69 78 79 96 8D 8D 8D 49 13 2E 14 AD	8D 4C 10 76 2A A7 18 8A 8A 90 17 16 8A 90 8D 61	9D 26 8D 23 DØ 10 AD 60 90 8D A9 22 90	90 96 97 90 80 80 40 80 40 80 90 64 21 90 09	A9 18 8A 78 A9 92 47 23 87 98 80 68	10<89> AD<56> AD<561> AD<561> AD<26> 00<85> 86<91> 69<65> 96<55> 98<84> BD<37> AD<26 20<89 20<85 20<85 20<85 20<85 20<85 20<85 20<85 20<85 20<85 20<85 20<85 20<85 20<85 20<85 20<85 20<85
85F8: 8D 86408: 8E 8610: 8D 8618: 4C 8620: 10 8628: AP 8638: AP 8638: 84 9640: 85 8640: 85 8650: 90 8650: 90 8660: C8	AA 90 0E 90 0E 47 ED 01 90 8D A9 97 20	8A 69 78 70 96 A9 8D 8D 8D 49 13 2E 14 AD 97	8D 4C 10 96 2A A9 18 8A 8A 90 90 8D 61 9B	9D 26 8D 23 D0 10 AD 60 90 8D A9 22 90 22	90 96 97 90 80 80 80 40 80 40 80 40 40 40 40 40 40 40 40 40 40 40 40 40	A9 18 8A 78 A9 A9 A9 A9 A9 BD A8 BB A8 PE	10<89> AD<56> AD<56> 60<61> AD<26> 90<85> 8A<91> 69<6E> 90<85> 90<84> BD<37> AP<68> AP<68> AP<69 AP<80
05F8: 0D 0600: EE 0610: 8D 0618: 4C 0620: 10 0628: 04 0628: 04 0638: 04 0640: 05 0648: FF 0650: 90 0658: 90 0668: 26 0640: EA	AA 90 0E 90 0E 47 6D 88 8D A9 97 20 98	8A 69 78 70 96 A9 8D 8D 8D 49 13 2E 14 AD 67 A9	8D 4C 10 96 2A 49 18 8A 8A 90 19 16 6A 90 8D 91 98 8D	90 26 80 23 00 10 40 90 90 80 49 22 90 28 48	96 AA 97 00 80 80 40 80 A9 64 21 90 67 97	A9 18 8A 98 A9 92 47 63 8D 68 98 CB	10<89> AD<56> AD<56> 60<61> AD<26> 90<85> BA<91> 69<6E) 70<84> BD <ab> BD<ab> BD<ab> AD&lt;37&gt; AP&lt;05&gt; AP&lt;05&gt; AP&lt;05 DD&lt;56 DD&lt;56 DD&lt;56 AB AP&lt;05 AP&lt;05</ab></ab></ab>
05F8: 0D 0600: EE 0610: 8D 0618: 4C 0620: 10 0628: 9 0638: 84 0640: 05 0648: F8 0658: 90 0668: 26 0668: 26 0668: 26 0678: 90	AA 90 0E 47 ED 81 80 A9 97 20 9E ED	8A 69 78 79 96 A9 8D 8D 8D 49 13 2E 14 AD 67 A9 07	8D 4C 10 96 0A A9 18 8A 00 19 16 0A 90 8D 01 9B 00 9B 00 9B	90 26 80 23 00 10 60 90 90 80 80 80 80 80 80 80 80 80 80 80 80 80	96 AA 97 00 00 00 00 00 00 00 00 00 00 00 00 00	A9 18 8A 78 A9 A9 47 03 02 87 90 80 80 68 90 68 90 80 80 80 80 80 80 80 80 80 80 80 80 80	10<89> AD<56> AD<561> AD<561> AD<26> 00<61> AD<26> 00<85> 8A<71> 69<65> 78<84> BD <ab> BD<ab> 20&lt;857  AP&lt;05&gt; 24&lt;80 20&lt;01&gt; 20&lt;55&gt; 20&lt;55&gt; 20&lt;55&gt; 20&lt;55&gt; 20&lt;56  20&lt;56  20&lt;56</ab></ab>
05F8: 0D 0600: EE 0610: BD 0610: BD 0618: 4C 0620: 10 0638: 04 0638: 04 0640: 05 0648: F0 0658: 90 0658: 90 0668: 26 0670: E8 0670: E8 0680: 90	AA 90 0E 47 ED 81 8D A9 97 20 9E ED 20	8A 69 78 69 70 8D 8D 8D 8D 13 2E 14 AD 97 A9 07 A9	8D 4C 10 96 4A 18 8A 00 17 16 8A 90 8D 01 98 90 97 99	90 26 90 23 09 10 40 90 90 80 49 22 48 49 20	96 4A 97 6D 8D 8E 4C 8D A9 64 21 99 97 95 53	A9 18 8A 78 A9 A9 47 63 62 87 80 80 80 80 80 80 80 80 80 80 80 80 80	10<89> AD<56> AD<561> AD<561> AD<26> 00<805> 88<91> 69<65> 96<55> 70<84> BD<37> AP<05> 20<80> 20<90 20<91 20<51 20<51 20<51 20<51 20<51 20<51 20<51 20<51 20<51 20<51
05F8: 0D 0600: EE 0610: 8D 0618: 4C 0620: 10 0628: 9 0638: 84 0640: 05 0648: F8 0658: 90 0668: 26 0668: 26 0668: 26 0678: 90	AA 90 0E 47 ED 81 80 A9 97 20 9E ED	8A 69 78 79 96 A9 8D 8D 8D 49 13 2E 14 AD 67 A9 07	8D 4C 10 96 0A A9 18 8A 00 19 16 0A 90 8D 01 9B 00 9B 00 9B	90 26 80 23 00 10 60 90 90 80 80 80 80 80 80 80 80 80 80 80 80 80	96 AA 97 00 00 00 00 00 00 00 00 00 00 00 00 00	A9 18 8A 78 A9 A9 47 03 02 87 90 80 80 68 90 68 90 80 80 80 80 80 80 80 80 80 80 80 80 80	10<89> AD<56> AD<56> 60<61> AD<26> 90<85> 8A<91> 69<6E> 90<84> 8D<37> AP<68> 20<80> 20<80 20<95> 20<95 20<95 20<56> 20<56> 20<56> 20<56 20<57 20<56 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57
05F8: 0D 0600: EE 0610: BD 0610: BD 0618: 4C 0620: 10 0638: 04 0638: 04 0640: 05 0648: F0 0658: 90 0658: 90 0668: 26 0670: E8 0670: E8 0680: 90	AA 90 0E 47 ED 81 8D A9 97 20 9E ED 20	8A 69 78 69 70 8D 8D 8D 8D 13 2E 14 AD 97 A9 07 A9	8D 4C 10 96 4A 18 8A 00 17 16 8A 90 8D 01 98 90 97 99	90 26 90 23 09 10 40 90 90 80 49 22 48 49 20	96 4A 97 6D 8D 8E 4C 8D A9 64 21 99 97 95 53	A9 18 8A 78 A9 A9 47 63 62 87 80 80 80 80 80 80 80 80 80 80 80 80 80	10<89> AD<56> AD<56<1> AD<56> 60<61> AD<26> 90<85> 86<91> 69<65> 96<55> 96<85> 20<80 20<80 20<80 20<81> 20<51> 20<55> 20<40 20<55 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20 20<56 20 20 20 20 20 20 20 20 20 20 20 20 20
85F8: 8D 86408: 8E 8610: 8D 8618: 4C 8620: 10 8628: AP 8638: 84 8640: 83 8648: FF 8650: 90 8660: C8 8668: 26 8668: 26 8668: 70: EA 9688: F0	AA 90 0E 90 0E 47 ED 80 8D 8D 49 97 20 98	8A 69 78 69 70 80 80 80 80 80 80 80 80 80 80 80 80 80	8D 4C 10 96 2A 18 8A 19 16 3A 90 8D 90 90 90 90 90 90 90 90 90 90 90 90 90	90 26 90 23 09 10 40 90 90 80 49 22 48 49 20 80	96 4A 97 6D 8D 8E 4C 8D A9 64 21 99 53 81	A9 18 8A 70 A9 90 47 23 62 80 80 80 80 80 80 80 80 80 80 80 80 80	10<89> AD<56> AD<56> 60<61> AD<26> 90<85> 8A<91> 69<6E> 90<84> 8D<37> AP<68> 20<80> 20<80 20<95> 20<95 20<95 20<56> 20<56> 20<56> 20<56 20<57 20<56 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57 20<57
85F8: 8D 8688: 8E 8618: 4C 8618: 4C 8628: 18 8628: 18 8638: 84 8648: FF 8658: 90 8668: FR 8668: 26 8678: 80 8688: 90 8688: 90	AA 70 0E 70 0E 47 6D 0E 8D 8D 8D 49 720 9B 5E	8A 69 78 70 96 8D 8D 8D 8D 49 13 2E 14 AD 97 A9 87 89 89 80 80 80 80 80 80 80 80 80 80 80 80 80	8D 4C 10 96 2A A9 18 8A 19 16 3A 90 8D 90 90 90 90 90 90 90 90 90 90 90 90 90	90 26 90 23 00 10 40 90 90 90 80 90 90 80 90 22 80 76	90 96 4A 97 00 8D 0E 4C 8D A9 64 210 9C 9F 91 83 81 18	A9 18 8A 70 A9 90 47 93 80 80 80 80 80 80 80 80 80 80 80 80 80	10<89> AD<56> AD<56<1> AD<56> 60<61> AD<26> 90<85> 86<91> 69<65> 96<55> 96<85> 20<80 20<80 20<80 20<81> 20<51> 20<55> 20<40 20<55 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20<56 20 20<56 20 20 20 20 20 20 20 20 20 20 20 20 20
85F8: 8D 8688: 9E 8618: 4C 8618: 4C 8628: 10 8628: 10 8638: 84 8648: 85 8648: 86 8658: 90 8668: 28 8668: 28 8678: 90 8688: 90 8688: 90 8688: 90 8698: E0	90 90 90 90 47 90 90 80 90 80 97 90 90 90 90 90 90 90 90 90 90 90 90 90	8A 69 78 70 96 8D 8D 8D 8D 8D 13 2E 14 AD 97 A9 8D 8D 8D 8D 8D 8D 8D 8D 8D 8D 8D 8D 8D	8D 4C 10 96 8A 8A 8A 8A 8A 8A 98 98 98 98 98 98 98 98 98 98 98 98 98	90 26 90 23 00 10 40 90 90 90 80 90 80 90 80 80 80 80 80 80 80 80 80 80 80 80 80	90 96 AA 97 00 80 00 64 21 90 97 91 18 53 81 18 81	A9 18 8A 78 A9 A9 A9 A9 A9 A9 A9 BD BD BD BD BD BD BD 99 BB BD 99 BB BD 99 BB BB 99 BB BB PB BB	10<89> AD<56> AD<56<1> AD<56> 40<61> AD<26> 90<85> 84<71> 69<65> 96<55> 70<84> 8D<37> AP<05> 20<80 20<91 20<91 20<51 20<51 20<54> AP<2E> AP<2E> AP<2E> AP<2E AP AP<2E AP
05F8: 0D 0600: EE 0610: 8D 0610: 8D 0618: 4C 0620: 10 0638: 04 0638: 04 0640: 05 0648: F0 0668: 90 0668: 26 0670: E8 0670: 90 0688: F0 0690: 90 0690: 90	90 0E 90 0E 47 6D 0B	8A 69 78 70 96 A9 8D 8D 8D 2E 14 A0 7 A9 62 09 A9 8D 8D 7 A9 8D 8D 8D 8D 8D 8D 8D 8D 8D 8D 8D 8D 8D	8D 4C 10 96 2A 18 8A 20 17 16 3A 90 90 90 90 90 90 90 90 90 90 90 90 90	90 26 80 23 09 10 60 90 80 90 80 80 22 90 80 80 80 80 80 80 80 80 80 80 80 80 80	90 96 AA 97 00 80 E 40 BA 98 64 21 90 C9 91 83 81 81 81 84	A9 18 8A 78 A9 92 47 23 22 87 80 80 80 99 99 99 99	10<89> AD<56> AD<561> AD<561> AD<26> 00<805> 8A<91> 69<655> 96<55> 70<84> BD <ab> BD<ab> 20&lt;805 20&lt;905 20&lt;901 20&lt;901 20&lt;51&gt; 20&lt;54&gt; AP&lt;2E&gt; AP&lt;2E&gt; AP&lt;2E&gt; AP&lt;2E 20&lt;54&gt; AP&lt;2E 20&lt;54&gt; AP&lt;2E 20&lt;54 AP&lt;2E 20&lt;54 AP&lt;2E 20&lt;54 AP&lt;2E 20&lt;554 AP 20&lt;554</ab></ab>
85F8: 8D 8688: 8E 8618: 4C 8618: 4C 8628: 19 8628: 94 8638: 84 9638: 84 9648: 57 8658: 90 8668: 26 8668: 26 8668: 26 8668: 90 9688: 70 9698: F0 9698: F0 9698: F0 9698: F0 9688: F0	90 0E 90 0E 47 ED 881 89 97 98 ED 98 98 98 98 98 98 98 98 98 98 98 98 98	8A 59 78 79 78 8D 8D 8D 8D 13 2E 14 AD 79 8D 8D 8D 8D 8D 8D 8D 8D 8D 8D 8D 8D 8D	8D 4C 16 96 8A A 97 8A 8A 98 98 98 98 98 98 98 98 98 98 98 98 98	90 26 90 23 09 10 69 99 99 80 49 29 80 76 80 76 80 76 80 76 80 76 80 76 80 76 80 76 80 76 80 76 80 76 80 76 80 76 80 80 80 80 80 80 80 80 80 80 80 80 80	96 AAA 97 60 A9	A9 18 8A 76 A9 P0 47 P0 43 8D 8D 68 P0 8D P0 90 P0	10<89> AD<56> AD<56<1> AD<56> AD<56  AD<26> 10<85  84<71> 69<65  96<55  98<84 BD <ab> BD<ab> BD<ab> 20<bd< 20="" 20<="" 20<bb="" 20<bd="" bb="" td=""></bd<></ab></ab></ab>
85F8: 8D 8680: 8E 8618: 8E 8618: 4C 8620: 10 8628: AP 8638: 84 8648: FF 8650: 90 8658: 90 8668: 26 8678: 80 9680: 90 9680: 90	AA 90 0E 90 0E 470 0E 470 8D 8D 477 20 9B 5E 8 9C 71 C7 A6	8A 59 70 96 A9 8D 8D A9 13 2E 14 AD 07 A9 02 09 68 00 00 00 00 00 00 00 00 00 00 00 00 00	8D 4C 10 96 2A A 18 8A 8A 90 17 16 8A 90 90 90 97 80 80 70 80 80 70 80 80 80 80 80 80 80 80 80 80 80 80 80	90 26 90 23 09 10 69 99 99 80 80 80 80 80 80 80 80 80 80 80 80 80	96 96 96 97 98 98 98 98 98 98 98 98 98 98 98	A9 18 8A 70 A9 A9 47 63 62 80 80 80 80 80 80 80 80 80 80 80 80 80	10<89> AD<56> AD<56> AD<56> AD<26> 90<85> 8A<91> AP<66> 96<85> 90<84> BD<37> AP<065> 2A<80> 20<408 20<408 20<56> 20<56> AP<225 20<56> AP<225 20<56> AP<225 20<56 AP<225 20<56 AP<25 AP 25
85F8: 8D 8688: 8E 8618: 8E 8618: 4C 8628: 84 8638: 84 8638: 84 8648: FF 8658: 90 8668: 26 8668: 26 8668: 70 8668:	90 0E 70 0E 470 8D	8A 69 78 70 96 8D 8D 8D 8D 8D 13 2E 14 AD 70 8D 8D 8D 99 8D 99 99 99 99 99 99 99 99 99 99 99 99 99	8D 4C 10 96 0A 18 8A 00 17 16 0A 98 00 98 00 17 18 00 98 00 99 00 18 00 90 00 18 00 00 00 00 00 00 00 00 00 00 00 00 00	90 26 90 23 0 10 60 90 90 90 90 90 90 90 90 90 90 90 90 90	96 96 97 98 98 98 98 98 98 98 98 98 98 98 98 98	A9 18 8A 70 A9 90 47 63 60 80 80 90 80 90 A9 90 A2 A9 80	10<89> AD<56> AD<561> AD<561> AD<26> 00<805> 8A<91> 69<655> 90<84> BD<37> AP<055> 20<80) 20<08> 20<08) 20<51> 20<51> AP<255 20<54> AP<255 AP<25 AP 25 AP 26 AP 27 AP 28
85F8: 8D 8680: 8E 8618: 4C 8618: 4C 8628: AP 8638: AP 8638: AP 8638: AP 8638: AP 8658: PB 8668: PB	AA 90 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	8A 69 78 79 8D 8D 8D 8D 49 12 14 AD 97 68 D0 A9 68 D0 A9 18 D0 A9 18 D0 A9 18 D0 A9 18 D0 A9 18 D0 A9 18 D0 A9 18 D0 D0 18 D0 10 D0 10 D0 10 D0 10 D0 10 D0 10 D0 10 D0 10 D0 10 D0 10 D0 10 D0	8D 4C 100 76 0A 18 8A 000 17 16 0A 79 8D 001 79 8D 001 79 8D 002 70 8D 002 70 8D 003 70 8D 003 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 8D 000 8D 8D 000 8 8 8 8	90 26 90 10 40 60 90 90 90 80 80 80 80 80 80 80 80 80 80 80 80 80	96 AA 97 08 0E 4C BD A9 0A 64 21 08 53 81 88 98 98 98 98 98 98 98	A9 18 8A 70 A9	10<89> AD<56> AD<56 AD<56 AD<56 AD<56 AD 50 A
85F8: 8D 8688: 8E 8618: 4C 8618: 4C 8628: 19 8638: 84 8648: 57 8638: 84 8648: 57 8658: 90 8658: 90 8668: 70 8668: 70 8668: 70 8668: 70 8698: 70 8698: 70 8698: 70 8688:	AA 90 0 6 90 0 6 7 0 0 6 7 0 0 6 7 0 0 6 7 0 0 6 7 0 0 6 1 0 6 1 0 7 0 0 6 1 0 7 0 0 6 1 0 7 0 0 6 1 0 7 0 0 6 1 0 7 0 0 6 1 0 7 0 0 6 1 0 7 0 0 6 1 0 7 0 0 6 1 0 7 0 0 6 1 0 7 0 0 6 1 0 7 0 0 7 0 0 6 1 0 0 7 0 0 6 1 0 0 7 0 0 0 7 0 0 0 7 0 0 0 7 0 0 0 7 0	8A 69 78 79 8D 8D 8D 8D 8D 7A 8D 8D 8D 8D 8D 8D 8D 8D 8D 8D 8D 8D 8D	8D 4C 100 96 A9 18 8A 200 19 8B 16 8A 90 8D 90 90 97 20 8C 20 20 20 20 20 20 20 20 20 20 20 20 20	90 26 90 10 40 90 90 90 90 80 80 80 80 80 80 80 80 80 80 80 80 80	96 AA 97 08 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	A9 18 8A 78 A9 79 47 23 8D 48 8D 48 8D 79 8D 8D 79 8D 8D 79 8D 8D 79 8D	10<88) AD<56> AD<561> AD<561 AD<26> 00<61> AD<26> 00<85> 8A<71> 69<65> 96<55> 98<84> BD <ab> BD<ab> 20<bd 20<55="" 20<bd=""> 24&lt;80 20<bd 20<55=""> 24&lt;80 20<bd 20<54="" 20<57="" 20<57<="" td=""></bd></bd></bd></ab></ab>
85F8: 8D 8680: 8E 8618: 4C 8618: 4C 8628: AP 8638: AP 8638: AP 8638: AP 8638: AP 8658: PB 8668: PB	AA 90 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	8A 69 78 79 8D 8D 8D 8D 49 12 14 AD 97 68 D0 A9 68 D0 A9 18 D0 A9 18 D0 A9 18 D0 A9 18 D0 A9 18 D0 A9 18 D0 A9 18 D0 D0 18 D0 10 D0 10 D0 10 D0 10 D0 10 D0 10 D0 10 D0 10 D0 10 D0 10 D0 10 D0	8D 4C 100 76 0A 18 8A 000 17 16 0A 79 8D 001 79 8D 001 79 8D 002 70 8D 002 70 8D 003 70 8D 003 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 000 8D 8D 000 8D 8D 000 8 8 8 8	90 26 90 10 40 60 90 90 90 80 80 80 80 80 80 80 80 80 80 80 80 80	96 AA 97 08 0E 4C BD A9 0A 64 21 08 53 81 88 98 98 98 98 98 98 98	A9 18 8A 70 A9	10<89> AD<56> AD<56> AD<56> AD<56> AD<26> BA<71> AP<6E> PA<8B> BD <ab> BD<ab> BD<ab> BD<ab 20<bb="" 20<bb<="" td=""></ab></ab></ab></ab>

```
86E8:F5 CE 8: 98 28 3F 9C A9<6C>
8AF8:81 8D 84 98
                  20 90 91
                            68<1A>
9AF8: 20
           98
              AD
                  00
                     90 C9
                            88(53)
        66
           24
0700: DO
        26
                  20 EA
        20 B4
               90
                        98
                           A9(EE)
0708:99
2718:22
        AR
           91
              CB
                  A9 88 BD 82(49)
                            42<3A>
                  83
                     90
0718:90 A9
           68
              BD
                         20
0720:9A
        28 61
               99
                  20
                     FE
                         98 A9(C2)
                     80
                        AE BRICEAD
0728:00
        8D
           98
               90
                  A9
0730:90
        9D 08 80 EE 00
                        90 AE(A2)
                         94
                            90(A2)
                  89
0738:00
        98
           AB
                     63
              20
        80
           EB
              CB
                  CB
                            F4<9F
0740:08
               49
                  04 80 04
                            98<3A>
0748:20
        $F
            90
0750:20
        90
           91
               AR
                  29 A5
                        9R ADCDD
0758:00
        90 09
               20
                  DØ 27
                         20 64(22)
                            A9(87)
0760:9B
        20 CD
               90
                  20 EA
                         98
0768:00
        AØ
            14
               91
                  CB
                     A9
                            BD(73)
0770:02
        98 A9 88
                  RD 83 98 28(58)
                            99(E6)
B778:42
        9A 28 AF
                  99
                     28 FC
0788: A9
        89
           80
              96
                  90
                     CE
                         20
                            99(5A)
0788: AE
        20
           98 AØ
                  00 89 63 9A(3D)
           63 E6 C8 C8 08 D0<1F>
08 9D 28 80 20 5F<76>
8790:9N
        28
0798:F4
        A9
87A0.90
        A9
           93 BD
                  84
                     90
                         28 9C(BE)
07A8:91
        60
           AD
              DL4
                  98 C9
                         83 F8KC2>
0780:26 C9
           24 FB 81 68
                         28 BB(F9)
0788:99
        A9
              AB
                  91
                     C8
                         A9
                            BBKFC)
           98
07C0:80 02
           98 A9
                  98 BD
                         83 98(1A)
07C8: 20 89
            99
              28 61 99
                         20 FBKF6>
0700:98 A9
           89 BD
                  21
                     90
                         68 28(06)
        99
           A9
                         91
@7D8: B7
               88
                            CB<17>
07E0: A9 90 8D 82
                  98 A9 88 80(D1)
        90
                  99
                     20
           20 89
                         F@
                            98(97)
07E8:03
07FB: A9
        99
            8D 01
                  98
                      60
07F8:90 C9 03 F6 05 C9
0800:23 60 20 98 18 99
                         84 FB(2F)
                         20 BE(60)
0808:99
         49
              AB
                      91
            98
                  14
                         CB
                            A9(2E)
0818:08 8D 82 90
0818:90 20 09 99
                  A9 88
                         BD 62<E4>
                  20 45
                         99 28(BA)
        98
0820:F0
           A9 08
                  8D @1
                         98
                            68KBD>
                            CB(A7)
0828:20 CC 99 A9
                         91
                  98 AB
0838:A9 08 BD 02
                  98 A9
                         28 BD<880
0838:93 90 20 09
                  99
                            99(29)
                      20 61
           99
0840:20
        45
              20
                  FØ
                     98
                         A9
                            08(FF)
                  AD 04
                         98
0848:8D 01
            98 68
                            C9(36)
0850:01 F0 05 C9
                  02 F0 13 60(5F)
        19
           9A A9
                            91(A2)
0858: 20
                     AØ
                         21
                  30
@860: CB
        20 FC
               98
                  A9 90
                         80 00<373
0858:98 68 28 88 99 A9 88 AB(6A)
0870:91 CB
           A9 88
                  BD 02
                         98 A9(80)
0878:08
        80
            23
               90
                  28
                            20(86)
0880:53 99 20 FC
                  98 A9 00 BD(14)
9888: 89 98 A9 AD
                  94 99 C9 B1(64)
0890:F0
        05
           C9
                  FB
                     23
                            20(FF)
               02
                         68
0898:20
        9A
           A9 00
                  A6 01
                         91
                            CB(21>
MRAM-AS MR RD M2
                  90 A9 00
                            ADCAD >
08A8: 83
        90
           26 42
                  9A
                     20 AF
                            99(E9)
9889: 20 FC
           98 A9
                     80
                  88
                            98(63)
08B8:68 20
           2E
               9A
                  A9
                     99
                         AB 91<4D>
RECE: CB 20 AF
               99
                  20
                      53
                         99 20(35)
08C9:FC
        98 A9
               98 8D 60
                         98 68(7E)
0800:A9 06
           8D
                     68 A9 81<13>
              88
                  70
0808: AM 00 91 CB A9 82 CB 91(F7)
08E0: CB 68 A9 BI
                  AØ
                         91 CB(86)
                     66
           AØ
                  91
20EE: A9 82
               14
                     CB
                            A2<3D>
08F0:00 AC 02 98 8D 2A 99 99(8D)
08F8:08 80 C8 E8 F0 08 D0 F4(56)
0900:A9
        88
           10
               99
                  17
                      90
                         AA
                            ACK 16>
0908:03
        98 99 88 88 CB EB E0<26>
0910:00 DO F7 68 NO F7
                         47 FE(1E)
0918:FF
         42 E7
               88
                  A9
                     3C 85 CB(21)
8920: A9 89 85 CC A9 00 A8 91<78>
0928:CB E6 CB
              28
                  FR
                     98 68 38(48)
2930: A5 CB E9 01
                  85 CB
                         A5 CC(DD)
0938:E9
        98
           85
              CC
                  60
                     18
                         A5
                            CB(83)
0940:69 01 85
              CB
                  AS CC
                         69 00(48)
0948:85 CC 60
              18 A5 CB 69
                            14(52)
0950:85
        CB
           AS CC
                  69
                         BS CC(CB)
                      88
095R: 40
        38 A5 CB
                  F9
                     14 85 CB(F5)
2960: A5 CC E9
              6965
                  85 CC
                         60 A9(D2)
0948:00
        8D 08
               92
                  8D 0C
                         98
                            BD<3AX
0970: 0D
        90
           8D 8E
                  99 68 A9
                            28<2F)
9978:BD 01
           98 AD 82
                     90 BD 07<D03
0980:90 BD
           ØB
              78
                  BD 89
                         90 80<762
         98
               18
                            90(CC)
8990:80 26 90 80 2D
                     98 80 2E<AD>
0978:70 60 AB 15 B1 CB 4C D0(41)
09A0:99 A0 01 B1
                  CB
                     4C DØ
                            99(54)
09AB: 26 45 99 A0 00 B1 CB AA<2F>
```

0980:20 53 99 4C E8 99 A0 13(93) 0988:B1 CB C9 03 F0 2F C9 04(F0) 0.9 89 FØ 21 C9 2F(83) 09C8: F0 2R 20 0900:7F 91 E0 03 F0 EØ. 04<12> 17 0908:FØ 13 EØ 89 FB 89 EØ 2EKCA> E9 09E0:F0 08 E0 4C(9A) 38 F0 60 68 89E8: C8 95 4C FØ 90 68 ADC 200 90 01 F0 83 4C 88(FE) 09F0:07 **C9** DOFR: 91 AR AR A9 01 8D 88 98(62) FB 94 9B 9A A0(6D) BA28: 4C 18 13 BA88: 14 21 CB 4C DØ 20 6F(FB) 61(82) 8A18:99 A8 00 B1 CB AA 20 BA18: 99 4C ER 99 20 AF 99 20(5F) 0A20:53 99 AB 26 B1 CB AA 28(44) 4C 0A28:61 99 E8 99(CA) 20 45 0A30:A2 00 02 90 SD 94(82) AC 8A38:99 88 88 C8 E8 E8 88 DØKASS 99<76> 8448:F4 A9 AA 88 AC 03 98 0A48: 08 90 CB EB EØ 98 00 F7<80> 18 5A 0A50:60 5A 7E 54 18 7E<63> 0658:5A A9 01 BD 237 92 AD 78(88) 0A60:02 C9 8E FØ **BC C9 OD** FØ<FB> 0A68: 20 A9 81 8D 08 90 4C FB<7F> 0070:90 09 00 8D 86 98 28 R2(39) 8A78:98 28 72 97 EE 86 ADKCB> 89:8HAG 90 C9 10 DØ FØ 4C FB(A7) 0488:9A A9 00 8D 86 98 28 82(21) 0A90:90 20 1A 97 EE 86 98 AD(E5) 98 8A98:86 **C9** 10 D0 FO 4C F8<81> 0000:9A A9 01 8D 07 90 AD 78<202 RAAB-WZ CO RB ER RC CO RZ FB(DS) BAB0: 20 BD 08 A9 01 98 4C FB(3E) 0AB8: 9A A9 88 ap 86 98 20 82(D1) 8AC8: 98 28 CD 96 EE 06 90 AD(08> 98 09 18 DO FR 4C **BACB: 86** FBKCFD 0AD0:9A A9 88 90 BD 86 20 82<D9> BADB: 98 28 81 96 EE 06 90 AD<79> ØAEØ: 86 98 C9 10 DO FO 4C FB(C7) BAEB: 9A A9 86 8D 87 98 68 ABKBE> 8AF8:68 68 4C DI 92 AD 88 984923 8AFB: C9 81 D8 @1 60 A9 30 AZKER) 9B @F 0000:89 20 14 9C 93 9B(A8) 0008:A9 64 20 83 9B A9 28KAA2 BC 0810:83 9B A9 84 20 83 98 60KBE> 0818:AD 08 98 C9 01 D0 01 ABKCE> ØB20: A9 85 A2 BA A9<95> 20 83 99 9B A9 8828:2D 20 83 20 83<34> 0830:98 A9 7D 20 83 98 60 ADC 7B> 80:8780 92 **C9** 01 D0 01 60 A9(88) ØB4Ø: CB 87 20 98 9B 9 CD<SE> 0848: 20 90 98 A9 CF 98 98(E9) 20 0850: A9 D1 28 98 98 AB AD RRCR53 8658:98 C9 61 DØ Ø1 A9 68 D5<35> 0860: AZ B9 28 90 98 A9 D7 20<ED> 0B68:90 9B A9 D9 20 90 9B 49(F9) 0870:DB 20 92 98 60 C5 CB DØ<95> 0878:16 E4 CC D0 12 68 68 40(56) Ø888: AB 9A C5 CB DØ Ø9 E4 CC<763 8888: DR 85 68 68 4C 83 9A ARK IDS 98 F& 0890:AD 0E 90 CD 1F @3<C5> @B98:10 20 60 AD ØD 98 CD 1E(D5) 28A8: 98 FB 03.3 10 15 60 AD DEKED> 00A8:90 CD 70 @A<4C> 1D FØ 03 10 0990:60 AD 0B 90 CD 10 10(92) RBBB: 01 AD AD AD 90 8D 1C 90<44> ØBCØ: AD ØC 70 8D 1D 98 AD RD<8A> 08C8: 90 ED 1E 90 AD ØE. 90 8D<CE> 0800:1F 90 AZ 98 AB 84 18 BD<88> ØBDB: 1C 9Ø 69 99 BC 8A 10 EB(BE) 09E0:88 E0 84 DB F1 68 3EKCC> 02<07> 0BE8:80 2F 02 A9 01 BD 6F @BF@: A9 A8 80 1.1 98 49 78 8D(2B) 98F8:00 D0 8D 90 10 A9 80 BD(63) 0C00:07 D4 10 9C **BB** 90 20 11<7C> 9C9B: 9D 20 4A 9C 49 98 AA 9D<27> 0C10:00 84 E8 D0 FA 20 38 90(5F) ØC18: A9 92 SD 1D DØ 99 E7<B3> 60 0C20:42 FF FF 42 E7 88 50 7E<87> 9C28:5A 1B 1B 5A Æ 5A AØ 80<B0> 0C30:AE 11 90 B9 28 9C 9D BOKERS

Listing zu »Crazy Scooter«. Bitte mit AMPEL eingeben. (Fortsetzung) MC38:84 EB CB CM MB DM E4 AMCCB> BC48:49 8D CO 02 A9 BC 8D<16> 4C ØC48: C4 BD C1 A9<88> OC50-70 RD. **C6** 02 60 ΔØ CHO 9D/52N 90 0C58:14 98 P.E 14 A9 88 80<89> 0C60:15 90 EE 15 90 ΑĎ 15 90(9F) RCAR: DR F8 4C D390 20 AD 904 090 OCTO- AD 1.4 90: CD GA 90 DØ F2(53) 0C78: A9 90 22 94(E2) BA BD ØA. 88 84<F9> **9080:40** 10 AD 84 DØ 09 ACAB-FR AS CO pes. EØ. 01 60 A9<38> 0C98: 88 SD. 203 D2 A9KCE> BD QI 1. D2 ØC98:01 24 90 20 29 96 20 (AE) OCAR: F1 94 20 74 95 20 93 EE 9EKFB> **RCAR: 28** 9h 20 31 ER 20 96<10> OCB0:98 20 2E 92 20 7C 9D 20<39> BCBB: DD 91 20 7D 99 20 1E 92<52> ACCA- 28 FO 90 20 73 90 AF 77(04) 90 90 BCC8: 98 AD C9<11> CE 13 13 0CD0:00 FØ 23 4C 77 90 AD 1B<AA> BCD8: 98 BD 13 98 4C 00 70 A9<31> 16<34> OCE 0:11 8D 05 86 A9 61 RD MCEB: 98 49 93 8D 21 90 A9 BA(92) OCFU: BD 18 90 04 80 28 98(4A) **A9** OCER: A9 20 an 20 98 60 49 77(27) 8098:85 C9 90 67 90(86) **9E** A7 OC. CA 40 A9 00 AA 9D 2D28:85 00<BF> 0010:B3 EB DB FA 49 BC: A2 22<7A> ES ER DØ F8(EB) QD18: 9D 20 83 **B**5 0020: A9 BD ØC. DØ 49 **C6** ADK 1F 21 85 00.28: 05 £500 68 CE 98 An 21(83) MD30:90 68 68 4C(27) C9 98 00 0038:B5 9C C9 92 FØ ØC. C9 01<0F> 9048:FØ BA 10 Δ9 AD. CA 4C<5E> 88 0048:5D 90 49 80 BD CC BA 40(01) 9D 0D50:5D A9 86 BD CB BA AD< 66> @F<F5> 0050:20 90 **C9** 01 ØE AD ADAM - OA AR 28 1E 92 68 gn DE COAN 4C 60 0D68:90 77 90 A9 8D<E8> BA 0070:CB BA BD CC BA 60 AØ 35<6B> 89 11 20 20 F1 DD78: A2 F 1 9n ABKEA2 **A9** 90 ADSA: 5F Δ9 OID. API. B9(24) 0D88: A9 29 Fi 90 AØ A9< 40> 83 28 90 AB 03 BA(05) 0D90:05 20 AZ 85 0D98: A9 20 F1 9D AR 29 ASCRE) PDAD: 09 A9 F1 MD(93) 9D AB 4F 28 75 @DA8: 20 9D AØ A9 11 20(39) 6080:F1 90 AØ 49 69 ØF 20 85<8A> 20 05 BBBB: 9F AZ 73 49 **BB** 9E<03> 9D DDCD: AN ARC410 A9 07 20 65 9E @DC8: C7 05 9E AB CB<88> A9 93 20 8008: 49 **PL3** 28 05 9F AR A5 A9(95) 8B<58> **7**F AB 47 0008107 20 05 9E 20(49) ODED: 20 05 9Ė AB 59 **A**9 PDEB: PS 9F 40 84 CB 86 CC BD(89) CB(62) **PDFD: 12** 90 AØ 88 **A7** 87 91 ODF8:C8 CC 90 DØ FB 60 B4(47) 12 0E00:CB SD 28 96 23 9F 12 98(16) OLOB: A9 89 85 CC A2 22 AB BB<BC> 0E101A9 91 20 61 79 EB(3F) 89 CB F3 A9 BF (A7) 0E10:EC 12 90 DO 60 7E<E7> 0E20:80 21 D2 20 73 9E 20 9E AD 70 C9(E4) ØF28:9E 89 24 28 66 11([3> FØ ØF A9 AB AE **BESB: 98** 90 88 84 EB CB CM Ø8< 71> ØE40: DØ F7 A9 88 AA 9D 08 62(88) **9**E 48: E8 EØ 10 DØ FB 20 58 9E (EC) 8E58: A9 86 BD 01 02 68 49 88KE5> @£ 58: AB C9<2D) 91 CB AB. 611 81 CB 0E50:82 DB 07 49 88 AØ 01 91(C2) @E68: CB 60 A9 88 AB 14 91 CB(B2) 8E78: 68 49 94 85 CF 49 9E 85<9F> 49 0E78:00 AC AC QF 90 85 CECCRY 0E80: A9 9E 4C AC A9(57) 85 DØ 9E 0E88: A4 85 CF A9 9E 85 4C<7B> NEGO: AC 9E 14(86) 20 88 20 200 50 0E98:04 910 52 CØ ØD **B**1 FIA 28(49) 42(65) **BEAB: 23** 80 26 54 82 81 98 REA8: 11 48 AD 24 90 **C**9 88 F6<C6> 90 OF(CC) AE. @EBØ: 12 AØ 86 11 **B1** REBB-9D AM BA ER CR CA R/A D8<533 AB B1 CF 99 0EC0: F5 88 28 88KF7> ØECB: CB CØ 88 00 F6 AØ 88 B1<D6> @ED0:CF 99 10 80 CB CØ D6<D6> REDR: FA 02 202 E8 00 (2)(2) AD BA<372 26E0:02 8D 20 D2 C8 D0 F7 EB< 75> A9 85 14<96> ØEE8:50 DØ FØ 68 80 14 C9 FA 06 FA 68 CE(BI) DEFB: 23 90 AD 23 90 C9 BB FB(F9)

11C8:00 00 00 00 00 00 00 00 00(EA) 1100:E3 80 FE BA DD OD. DO DOCAES 11DB: 00 212 02 20 aa PRICEAS 20 20 11E0:00 ØЙ 89 11FR: 00 20 88 DIGI GIDI PM OU. DOK DOX 11F@:00 CAPI PLA. 29 48 BA 4B<33> 86 11F8: A6 BD D4KE5X 1200:80 DR FA. D1 AB ARCRAN 16 00 1208:40 7C 7A 74 7E 78 76 76<99> 1210: 7B 70 7E 78 78 76 76(28) 1218:78 78 7C 7E RR BA RC. 38(82) 70 590 FR 1228: 3B RØ. an DIG. PB<245 1228:00 818 89 80 70 88 Ė7 42(5C) 00 88(4A) 1230:FF FF 42 £.7 (AR) 66 1239-00 0.0 DICH Dig. 20 ChOL DUD DURK SCO 1240:FF FE 16 18 18<48> 97 690 20 1248: 1B 18 18 18 18 20 88<A3> 1250: 1F 1F 18 18 18 18 18 18<31> 125B: 1F 1E 88 (36) 00 RIO. (3(7) 88K53> 1760: FB FB 19 18 18 18 18 18<88> 1268:F8 F8 80 88 80 20 PARKERS 88 88 50 1270:18 18 88 0.0 7E 50(A3) 1278:18 5A 7E 80 30 99(92) 1288: AL AL 99 42 30 18 18 18<5F> 1288:00 88 88 88 80<00> (30) 82 86 1290:18 30 60 40 @C<19> 99 99 18 1298: FE **OC** 18 89 202 89(A1) 28 98 1248:80 88 66 20 2021 ZC. CA CACRES 12AB: C6 C6 Çá 70 38 98 18 18<9A> 18 18 7E 1290:18 22 7C Cá **9C**<4B> 68 63 1288:18 30 FE (AG) 70 CA. .06<44> 96 1208:00 70 3C<91> 88 OC. 10 12C8: 6C CC FE 9C 80 98 FE BOK BE) 86 C6 1200:FC 7C 20 7C C0<22> **C6** 12D8:FC 7C Cá C6 20 FE. 86 **BCKEC** 12EØ: 18 30 20 30 7C Cá C6<48> 80 7C 12E8:70 C6 **C6** 7C 90 C6 C6<C1> 00(45) 12F@+ 7F 86 CA 70 00 (200) OHO 12F8:00 99 88 88 99 20 00<1C> 00 1300:00 88 90 00 00 88 22 00(26) 1300:00 98 20 22 20 88 20 00<2E> 98 1310:00 20 PIO. 00 OLO (2)(2) 00<36> 99(3E) 1318:00 88 22 20 28 22 20 1320:00 FC 00 80 D7 81 99 80KCE) 1328: 88 88 88 PHP DO 00 DO. 100<4E> 1330:00 FE E2 BE FE 82 82<77> 82 1338:00 FC E6 FC BE FC(DB) 1340:00 FE FO 80 BO 90. FØ FE(CB) 1348:00 FB EC FB(39) CA 82 CA EC. 80 1350:00 FE FØ FR 82 FO FE(57) 1359:00 FE F0 80 F0 88 80 B0<F0> FØ 1360:00 FE 80 9E 82 F2 FEKEA> 368: 88 C6 C6 82 FE CA C6<3C> 82 1370:00 FC 30 30 30 360 FC(35) 30 1378:00 86 86 26 42 42 66 30(A1) CC 78 1380:00 C6 CC 98 FØ C6<56> CØ 80 1388:00 CO CØ FE(AF) 80 BE 1390:00 82 C6 EE BA 92 **C**6 C6<1B> 1398:00 E6 C6 92 DA **B6** CE C6(A7) C2 C6 FE(60) 1300:00 FE DE 86 F6 1348:00 FE E2 8E FE 80 FIRM 80(D0) 1380:00 FE 82 82 82 BE BE FE < ØA> 1388:00 FE 38 FE 98 80 E2 86<5E> 1209:00 FE FØ 80 FE FEK9EX 02 13C9:00 EF SB 3B 30 10 10 18(2B) 1300:00 82 82 82 C6 C<sub>6</sub> C6 FE(7E) 1308:00 C6 C6 82 Cá 6C 38 10(A5) 92 EE C6 82<34> 1360:00 C6 **C6** BA 13F8:00 CA EE ZC: **3B** 70 EE CACRAS 1350:00 CC CC 48 7B 30<13> 30 30 13F8:00 FE C6 ØC. 38 68 C6 FE(DD) 1400:00 50 816 56 86 49 (NO) RSCAAS 1428: D1 4C 62 E4 70(96> 700 ZC. FR 1410:AD 18 98 C9 @1 FØ 1E C9(E8) 1418:02 FØ 10 **C9 B**3 FØ 10 C9(76) 1420:04 FO 18 AD 16 98 C9 01(F9) 1428:F0 29 09 69 92 FB 5.3 04<2D> 1430:F0 4C 67 70 40 CE 7D<24> 1438:4C FB 7D 4C FC 7E 21(85) 4C 9R 20 1440: 7F 28 70 16 7E FECOR) 1448-18 90 90 AD **BD** 10 200 DØ4 795

Listing zu »Crazy Scooter«. Bitte mit AMPEL eingeben. (Fortsetzung) 1450:6C C9 00 20 C2 7C 20 6D<0F> 1458:70 CE 18 98 AD 18 90 BEX RE> 1460:00 DM 6C C9 00 20 E9 70(85) 7C EE 20 79 11<85> 1468: 28 54 Æ 1470:90 20 **B**9 6C C9 28 1478:10 7D 20 98 7E 28 79 89 7C 6C 7C<1E> 20 89 98 A9 C9<9D> 1480: CF 11 99 1488:00 AE 11 99 AB 9D< 6A> 1490:00 84 E8 CB C6 48 D8 F7<07> 1498:40 AE 11 98 A8 88 B9 64K4B> 90 00 84 E8 CB C0 14A0:7F **BB**<8F> 14A8: DØ F4 68 A9 03 8D 16 98(46) AB 20 7D 1480:AZ BB AD 37 AZ<7E> 1488:A8 A0 98 20 37 7D A2 98<883> 7D A2 88 14C0:A0 88 70 37 ABK 25> 14CB: 78 20 7D **E**9 82 80 37 16(E1) 04 BD 1400:90 68 A9 16 90 A2KAC> 14D8:38 AØ 28 20 37 7D A2 49(47) 70 A2 58 14E0: A0 58 20 37 AØK 9B> 14E8: 48 20 37 70 A2 68 A8 58(24) 7D A9 01 8D 14F0: 20 37 98<8A> 14F8:60 A9 02 BD 16 90 A2 1500:A0 AB 20 37 7D A2 48 38<33> ABC7E> 1508:98 20 37 70 FC 7C 7DK 310 1510:A2 58 A0 88 20 37 1518:68 A0 78 28 37 7D 7D A2<36> 7D 49 844405 98 68 A9 81 16(86) 1529: BD 16 8D AZ BB AØ 7D<40> 1528:90 28 20 37 1530: A2 A8 A8 38 20 37 7D 1538: 98 A8 48 20 37 7D A2 1540: A8 58 28 37 7D A9 03 A2(87) 88<88> BD(A1) 28(AE) 98 68 EC 10 90 00 1548:16 1550:CC 11 90 D0 13 68 68 68<66> C9 1558: 68 AD 16 98 01 F0 BAKEIS 86 1560:C7 02 FQ 20 7C 00 1568:00 7C 20 46 7F 4C 70(83) 1578:68 AD 19 98 C9 81 FB 81<7D> 1578:60 A9 00 00 19 90 68 68<45> 1588:60 20 5D 7D A9 78 CD 10<23> 98(94) 49 28 CD 1588:90 D0 0F 11 44 7F 1590: D8 88 20 C9 1598:5F A9 78 CD 18 98 DØ<BC> 68 15AM: WE A9 38 CD 11 99 D0 **BB**(EA) 7F C9 81 FB 27 15A8: 28 44 68<C6> 78 1580:A2 38 28 AB 7D 1588: 78 A@ 48 28 AB 70 60 EC(7D> 15CB: 18 98 D0 11 CC 11 90 D8<75> 15CB: 0C 68 68 **C9** 61(88) 28 44 7 15D0:FØ Ø4 40 E4 7D 68 68 48<50> 1508: A9 Ø1 aD 18 98 49 88 BD(DE) 70 28 70 15E0:17 79 7C CE 11 98<8A> 15E8: 20 89 EE 17 99 AD 17(16) **C9** C9 18 FØ 25 15F0:90 6C 68<FF> 49 15F8: 68 68 02 BD 18 90 A9<34> 1600:00 SD 17 90 79 7¢ EE(89) 28 1608:11 90 20 89 FB 70 F3 7E(56) EE 17 1618:7C 98 AD 17 99 C9(B3) C9 1618:19 FB 6C 88(87) 1620:80 18 98 20 4A 7F 89 81(28) 90 AC E9 1A28: 8D 19 010 28 50(90) A9 78 CD 98 DIS @F<12> 1638:70 18 1638: A9 AB CD 1.2 98 DB 68 28(59) 1 /449 - 44 7E **C9** 01 FB 94 AR 09(45) CD Dea OF. 164B: 7B 12 90 89 78(F2) 98 08 88 1650: CD 11 28 44 A0 60 A2 7D A2 78 1658:09 81 FØ 78 ARK BES 1668:98 28 AB AB BR(FB) 7D 60 1668: 20 AB 29 270 A9<00> 50 68<5A> 1678: 38 CD 98 DØ ØF 10 11 1678:CD 90 00 08 20 44 7F<16> 1680:C9 @1 FØ 29 68 A9 ABI CD<17> 90 DØ ØF 1688:10 68 CD 11<3D> 1690:90 00 28 28 44 7E BIKAE) A2 48 A8 68 169B:F@ 10 68 28(52)

16A0: D6 7E A2 58 A0 68 20 06<70> 7E<D3> 1668: 7E 60 4C 12 /F 4C ED 16RB: 20 SD 70 A9 88 18 90(F3) **QF** A9 68 CD 11 98 DB(C7) 1688:00 1600:00 20 44 7F C9 01 F0 A2KED> 88 CD 10 98 D0 16CH: 69 A9 DF<1E> 16D0:A9 68 CD 11 90 00 89 20<01> 27 16DB: 44 **7F C9** 01 FØ 68 16E8:98 AØ 68 20 D6 7E. A2 A8<E3> 1AE8: A0 AB 20 DA 7E 6B EC 10<21> 16F8: 90 D8 11 CC 70 DØ BC(A9) 11 28 44 7F **C9** B1 FB(CB) 16FB: 68 68 4C 12 7F 1799: 84 68 88 1708:03 8D 18 90 F4 7E **6B** 7F(C2) 1710:20 4A 7F A9 88 8D 17 90(5B) 98 AD 18 90 1718: EE 10 an. 20KC7> 1728:DØ EE 17 99 AD 17 98 C9<47> 28 &C **6**B 68(95) 1728:18 FB C9 1730:A9 04 80 18 98 20 46 7F<05> 1738: A9 00 0D 98 CE 17 18 98(21) 88 1740:AD 18 98 Sp DØ 17(05) EE 1748: 90 AD 17 98 83(AE) **C9** 10 F0 1750: AC C9 RR A9 on on PID. 18 98(F7) 1758:A9 01 8D 90 19 00<7A> 6C **C9** 1768: AD 8A 02 29 B1 68 AØ 1768:AE 11 98 B9 50 7F 90 88KCF> 1770:84 E8 C8 C0 08 D0 F4 ARKAS) E7 42 FF 42 1778:00 E7 FF 00<67> 5A 18 5A 7E 18 1788:E0 02 E1 02 40 988 88 88(A7) Laenge 6030 Bytes

Listing zu »Crazy Scooter«, Bitte mit AMPEL eIngeben, (Schluß)

### Pacman mal 2

Versuchen Sie sich gegen pfiffige Monster. Schlagen Sie sich durch Gänge und sammeln Sie Punkte, alleine oder zu zweit.

as macht mehr Spaß als Programmieren? Natürlich ein erhotsames Spielchen. Und was macht noch mehr Spaß! Wenn zwei Personen gleichzeitig gegeneinander antreten. Bei »Twomaze« handelt es sich um ein Spiel für zwel Personen. Es ist in Maschinensprache geschrieben, verfügt über hervorragende Grafik und ist besonders schnell.

Twomaze« ist eine Pacman-Variante. Wie auch beim Original, besteht Ihre Aufgabe darin, möglichst viele Punkte zu fressen. Aber Vorsicht! Sie sind nicht der einzige, der sich in den Labyrinthgängen bewegt. Kleine, gierige Monster trachten Ihnen nach dem Leben. Allerdings gibt es noch Kraftpillen, die Sie besonders dann nicht verschmähen sollten, wenn sich Ihnen einer oder mehrere der Bösewichte nähern. Haben Sie rechtzeitig eine Kraftpille verspeist, verfügen Sie über enorme Kräfte und können Ihre Gegner vernichten. Achten Sie dabei aber auf die Zeituhr am rechten Bildschirmrand! Wenn diese nämlich abgelaufen ist, sind Sie wieder ein normaler Pacman und müssen sich mit äußerster Vorsicht durch die Gänge bewegen.

Für das rasante Spiel ist übrigens mindestens ein Joystick nötig. Über die Tastatur lassen sich die Pacmen nicht steuern. Wer glaubt, daß der gute alte Pacman schon zum alten Eisen zählt, hat sich getäuscht. Bei unserem Listing »Two-maze« gehen nämlich gleich zwei »Pacmänner« gleichzeitig auf Punktejagd (Mampf!). Schnappen Sie sich einen Freund und zwei Joysticks und schon kann die Labyrinth-Hatz mit einer aufregenden, neuen Spiel-Variante losgehen. Pacman-Spieler aller Länder vereinigt euch und tretet in diesem furiosen Simultan-Programm gegeneinander an! Das ideale Spiel für Turniere und Punktspiel-Betrieb (Gobbel-Gobbel-Schluck!).

Zur Eingabe von »Twomaze« verwenden Sie bitte AMPEL Beachten Sie auch die Hinweise zu diesem Eingabeprogramm. »Twomaze« läßt sich übrigens nicht von Basic aus eintippen. Dies gilt für alle abgedruckten AMPEL-Listings. Und jetzt noch viel Vergnügen bei der Monsterjagd.

(Frank Ostrowski/wb)

PROGRAMM-STECKBRIEF						
Programmname	Twomaze					
Programmtyp	Spiel					
Programmiersprache	Maschinensprache					
Programmlange	4219 Byte					
für Computer	alle					
zusätzliche Hardware	Diskettenlaufwerk oder Kassetten- rekorder, ein oder zwei Joysticks					
Eingabehilfe	AMPEL					
Bemerkung	Pacman-Variante für zwei Personen					
Leserservice	Diskette (TWOMAZE.COM)					

0000:FF FI	F EB	02	Εı	92	86	20<16>	1	82C8: A3	CA	10	FB	AØ	4F	89	£7 <b2></b2>
999B: 99 21	2 6E	249	70	21	80	20<75>		102C8: 22	99	99	22	88	10	F7	20 <b1></b1>
0010:06 20	0 A9	3C	8D	82	D3	A9(AF>		0200:1C	2F	68	188	00	99	98	88<7A>
0018:06 8		A9	20	85	ØD	A9<62>		0208:00	98	99	98	88	20	88	@@ <dc></dc>
0020:01 8:		A9	98	80	44	Ø2<3C>		B2E6:00	99	90	2C	25	36	25	20<17>
0028:40 FI		78	60	42	00	32<33>		02E8:00	92	10	28	96	88	20	60 <ee></ee>
2030:02 20		DE	21	02	02	02(59)		02F9:00	20	88	98	99	98	96	808 <f4></f4>
0038:02 0		88	47	82	20	10(09)		02F8:00	98	99	88	98	90	26	00 <fc></fc>
0040:42 E		96	44	88	39	04(51)		9399:00	10	20	99	1212	90	88	00 (0A)
0048:04 04		84	24	84	24	Ø4<48>		0308:00	28	29	27	28	99	98	90(EB)
9258:04 R4 9258:60 43		32	41 02	1F AØ	20	78<4A>		0310:00	92	88	98	10	22	98	00<96>
0058:50 4: 0058:38 04		04	04	24	04	28(FA) 84(7A)		0318:00 0320:00	10	20	27	21	20	25	00<1E>
0068:04 B		24	04	04	64	04<68>		0328: 2F	36	25	32	20	27	23	20<1E>
0070:04 0		24	04	24	64	41 <ad></ad>		0330:68	24	60	49	88	A2	04	95(20)
0078:4B 2		EE	E5	88	70	6C <d4></d4>			CA	10	FB	85	AE	85	AF (C9)
0080:41 79		72	88	4D	90	20<44>		0340:85	9E		9F	20	10	20	A9(FD>
0088:E5 E		59	88	96	F4	F7(80)		8348:28	85	80	A9	38	85	61	A2 <a7></a7>
0090:EF 00		6C	61	79	65	72(04)		0350:17	AB	26	A9	80	91	88	88KEA>
0078:00 4	D 88	20	E5	EL	F3	F9<75>		9358:18	FB	18	A5	80	69	28	B5 <ed></ed>
00A9:00 0	Ø EF	EE	E.5	88	78	6E <f1></f1>		0360:80	A5	81	69	82	85	81	CAKE5>
00A8:61 7	9 65	72	88	40	20	EE(BA)		8368:00	E7	9	F3	85	88	A9	39<06>
2030; EF F	2 ED	Ea	EC	99	F4	F7<96>		8378:85	81	A9	48	BD	F3	39	A9<35>
00B8:EF 0	0 70	6C	61	79	65	72<1C>		0378:00	ap	51	38	8D	75	38	AD(A2)
00C0:00 4	0 98	EE	EF	F2	ED	E1 <ed></ed>		P8:0878	D2	29	92	85	82	85	83<2F>
QUCB: EC Q	8 EF	EE	E	66	70	&CKEF>		2388:2A	AB	A5	89	79	23	25	85<81>
00D0:61 7		72	98	4D	98	ØØ<34>		8376:84		81	79	24	25	85	BS< ØD>
OODB: EB E		E4	20	60	F'4	F7<60>			20	B1	84	CP	88	DØ	29<5A>
ØZEZ:EF Ø		4C	61	79	65	72<30>	-	03A0:A5	82	18	69	@1	91	84	69<1D>
22E8:20 41		22	£8	E1	F2	E44067	1	83A8:83	88	AB	A5	196	79	23	25<45>
00F0:00 0		86	98	88	20	図例(F図) 図例(F図)		9269:85		A5	81	79	24	25	85<19>
00FB100 3		6C	65	63	74			0388:81	A9	99	AB	91	80	A5	84<79>
0100:27 6: 0100:72 6:		65 73	20	34	72	30<46> 69<30>		03CD: 85	80	A5	95 A5	85 82	81	4C	73<57>
Ø11Ø:67 6		72	88	88	86	00<43>		0300:23	C5	63	De	B3	AB	93	85<98>
Ø118:00 0		00	4C	4C	4C	00444>		23D8.82	63	82	98	91	88	24	82<2E>
0120:4C 0		20	4C	20	40	4C-8F>			18	C6	82	A5	82	BA.	ABK 7C>
0128:4C 4		4C	4C	4C	40	4C<9E>		82E8: 28	A5	98	F9	23	25	85	80 <aa></aa>
2139:22 40		4C	4C	99	4C	4C<88>		03F 0: A5	81	F9	24	25	85	81	4C<99>
0138:4C 4		40	4C	4C	4C	00<62>		63F6173	23	86	A1	BD	43	25	85<71>
0140:00 0		20	20	4C	20	00<73>	1	0400:95	AD	OA.	02	29	3F	C9	25(99)
@148:4C @	8 88	00	4C	92	40	00<6B>		8498: BØ	F7	AB	C8	84	97	AD	BA<93>
0150:00 4	C 00	4C	99	4C	88	4C <a7></a7>		0416:DZ	29	1F	C9	15	90	F7	AAKBE>
0158:00 4	C 98	98	4C	98	98	00 <cf></cf>		0410:45	97	29	01	FØ	E3	90	18<30>
0160:4C 40		4C	50	90	99	CO.DF>		0420:69	D6	82	88	A9	37	69	@@ <df></df>
@148:00 0		20	88	4C	62	88<5B>		Ø428:85	81	18	A5	80	69	29	85 <df></df>
0170:40 0		90	40	90	4C	ØØ < 7A >		04 10:80	A5	81	69	90	85	81	CA< 4B>
0178:20 4		4C 4C	90 40	4C	88	4C17F>			FØ	AØ	52	BI	80	FØ	C1<81>
0180:00 40 0189:40 00		4C	4C	4C	99	4C(82) 00(05)		0440:AZ	03	RC	33	25	B1	86	F@<61>
8190:00 8		26	20	4C	200	88 (C3 >		0448:11 0450:25	11	37	25 BC	3F	30	BC 11	38<38> 88<29>
0198:4C 0		98	4C	88	40	BØ3 45>		0458:FØ	A7	CA	10	E5	25 A9	80	AB <c6></c6>
81A9:00 4		4E	98	90	80	4C(C6)		0450:52	91	30	C6	95	DØ	98	BC(6C)
01A8:00 4		88	4C	86	4C	4C <fc></fc>		946B: 52	38	BC	79	38	SC	74	38<51>
0120:00 0	-	4C	86	86	80	ØØ< 77>		9470:8C	9D	38	68	A9	EF	85	BØ<34>
01BB100 0	88 9	88	28	4C	98	283>86		0478: A7	37	65	81	AR	29	B1	BØ (AE)
01C0:4C 40	C 4C	40	4C	88	4C	4C<71>		6480:F0	2E	A2	50	40	01	D1	88(29)
Ø108:40 40	C 00	40	66	90	99	4C <d4></d4>		8488:FØ	Ø1	E8	AD	20	B1	80	F8KCE>
@1D@:@@ 40	C 88	66	4C	26	4C	40<24>		0490:02	EΒ	EB	AØ	51	81	90	F@ <da></da>
21D8:4C 40	C 98	4C	4C	4C	4C	<b>86</b> <82>		0498:04	€.8	E8	EØ	E8	AØ	28	BI (DD)
0150:00 0								94A2:88					-		-
01EB: 00 0								84A8: E8						91	
01F0:00 2							1	0480: E6	-					_	
Ø1F8:72 0								8480:C9							
8299198 9								84C0: 00							
0208:00 99 0210:87 A								Ø4C8:85							80<94>
0218:20 4					_		1	0400:08 0408:45							9F<50>
0220:28 A															DØ<92>
0228:20 6								04E0:41							86<82>
0230:2F 2								04FØ: A5							
0238:85 B								04FB: D3							
0240:20 5								2522:71							
0248:27 4			_					0508:8D							
0250:04 A								0510:85							9F < D4 >
0258:3C B								Ø518: A7							38<92>
0260:18 2								8520:A9			52				38<90>
026B:99 0								0528:8D							@2<6B>
0270:8D F	4 82	68	A9	30	BD	07<22>		0530:00	FE	FF	50	98	BØ	E.E.	Ø1<5C>
0278:D4 A								9538:98			28	88	DB	FF	ZAKEF >
0280:BD 2								8549:51							29<00>
9288:A9 9								0548:50							
0290:35 9								0550:1C							
8298:08 3								0558: 0E							
02A0:B9 9								0560:86							
Ø2A8: F7 64								0568:00							DØ(FF)
0280:CA DI								0570:0E							
Ø2B8:48 4	7 411	149	1010	HZ	03	73(28)	4	0578:AZ	IONG	86	40	-210	/E	26	MUSCO>

0580:8A 29 87 C9 84 F0 08 A5<1C> 0588:8F 29 0F 0590:8A A4 8F CA 98 DØ BE A6<54> 29 20 27 A2 B4<F3> 7E 2598:86 95 28 26 AS. 95<97> 86 8A<9C> 05A0:95 E6 96 A6 95 85 86 @5A8:15 8B 29 @F DØ Ø3 20 CA(AA) 95 85 05E0:25 A6 AB DØ 16 B4<313 0588:90 A5 96 40 BØ 87 85 #500:79 17 26 95 86 18 B5 88<73> 05CB: 79 1D 95 88 C6 95 10(9C) 26 0500:D2 60 95 BD **2B** 85<48> 05D9:97 84 88 85 86 AA 28 ZA(F4) 05E0:27 AZ FF 99 A2 D3 BC(3C) 86 80 ØSE8: 23 26 **B**1 **C**5 97 90 82<8F> Ø5F0: A9 00 95 9A BB 82 E6 99475> 95FB: CA 10 EC A6 95 FB 44 FB(16) 32 0689:04 FB A5 99 32 18 F8<5A> 0608:0A B5 90 49 01 AB A9 01(E9) 0610:79 88 AD 8A D2 7A 29 83(85) F4 6618:AB AB **B9** 94 38 DØ 98(F2) 0620:95 90 60 01 FF 88 86 00<CE> 8628:88 68 99 91 FF 88 88 28(66) 8638:28 51 81 29 29 24 83 30(BE) 0638:E9 AD 20 D3 40 40 4A(67) Ø640:4C 48 26 24 81 30 DB ADKFAX 29 ØF 85 97 Ø648: ØØ D3 24 87<9C> A2 Ø1 46 97 0650:38 D0 80 07(BF) 0658:A5 9D 90 23 AB 23 CA 8668:97 88 87 A5 90 D8 83 A8(A1) 0668:02 CA 97 98(80) 88 87 46 A5 9670:00 93 AB B1 97 CA BB(93) 8678:07 A5 9A D8 03 A8 00 CA<02> 0680:8A F0 02 A0 94 A6 95 98(85) Ø688:95 98 86 95 AØ 29 81 88KEC> 0690:C9 41 F0 39 C9 40 F0 32<2D> 0698:C9 42 F0 01 60 20 BE 0680:A9 16 85 AE A9 07 85 26(04) AF<33> 0648:42 00 BD FC 26 85 80 EB(29) MARO: RD FC 26 85 81 AØ 33 A9< B7> 80 E8 EB 2C D8 E4<03> 8698:4A 91 06C0:A9 00 A2 84 95 AB CA 10<90> 0608:FB 60 A9 01 0600:AD 20 DF 26 20 A9 BA 86(49) 26 A9 20 AB 29(2F) Ø6D8:91 80 A9 61 65 86 A5<72> 48 06E0:9E D0 02 C6 9F 66 9E 68KFD> 06E8:60 A9 01 E0 00 F0 02 A2(52) 06F0:02 AC 08 D4 C0 F9(AD> 94 90 06F8:00 7B **BØ F**5 19 75 A3 95(09) 6700: A3 85 A4 69 00 95 A4 ARC175 94 0788:1C 38 44 38 6C 38 38<DB> 0710:BC 28 £4 30 2C 39 0718:5C 0720:FC 39 84 39 AC 39 24 3A 4C 39 D4 39(E9) 3A 74 3A(FE) 0728:9C 3A C4 3A EC 3A 14 38<5A> 0730:30 3B 64 3B 8C 3B 98 4A<33> 0738:4A A8 8A 4A 4A 18 79 FB(BE) 0740:26 95 80 99 F1 26 69 00<21> 0748:85 81 60 A5 9E 85 9F D0<4F> 0750:1C 20 05 20 20 20 23 20(A7) 0750:62 27 A6 AD A9 64 20 DE < 7E > 8768:26 A5 A3 85 A4 D8 FA A5<28> 0768:A5 05 A6 D0 F4 60 A6 B6(FA) 0770:A9 00 85 BA BD 27 79 95<B9> 7E 27 95 88 0778:86 BD CA 10KD3> 0780:F3 8E 1E DØ 60 30 30 7B< 04> 0798:00 00 30 00 00 00 30 A5KAF> 8790:A1 8A AA 9D 24 28 AB BD(C1) 0798:25 28 AA 98 DØ 91 CA 88<71> PIZARIER FE DR F7 A5 AF MS AFCARS 0748:F0 42 A5 96 29 01 D0 3C(66) 0780: A5 AE BA AA BD 26 85(47) 0788:00 BD FD 26 85 81 A9 00<75> 217 C.O : ARI 91 88 AA AF A4 AF(AF) 33 07CB:88 10 93 AØ 07 CA 86 AE(13) 0700:84 AF 8A 00 04 CØ 00 F0<A9> 0708:13 0A AA BD FC 26 85 80<F3> 07E0: BD ED 26 85 B1 B9 A7 22(09) 91 80 87E8: A8 C9<88> 33 AD FC 02 07F0:21 00 13 A9 FF GB FC 02(24) 97F8:20 FA 27 A9 FF CD FC 02<29>8800:F0 F6 8D FC 02 60 AD 2A<09> 2828:32 C5 82 D0 05 A2 99 20(84) 0810:C1 2F AD 4A 32 C5 84 DØK5E.> Ø818: Ø5 A2 Ø2 20 0.1 75 09 RRICAFO 1F DØ DØ 0828:A9 01 20 68 20(34) 0828:1F D0 F0 FB 4C 10 22 68<20> 0830:D0 07 82 06 94 05 E2 Ø4<1B> 0838:57 04 E8 03 80 03 41 03(20)

Listing zu »Twomaze«. Bitte mit AMPEL eingeben.



2840:01 03 CA 02 9A 02 71 02<AA> 2B 82 8E 82 F4 01<C5> 28 AØ 20 5CK3B> @850:A9 07 **A2** 5D MASA-FA 49 20 NO OR 92 A9 28(57) D4KEB> 02 A9 SD DE 0860: BD 01 CO 0868:60 28 3B 28 20 DD 20K2F> ZA 28 72 28 20 4F 2B(37) 0878:20 EB **7F** 4C 62 E4 A5 14×F7> A3 A4 FREDE 14 AS 25 @880:4A DEE Ø888:31 48 **A3** 02 C6 A4(99) A3 68 0890:C6 AØ 2D DØ 12 A5KE5> A5 A5 (6A) 8898: A5 925 86 EØ. 10 46 A6 **A5** 68 ABK 6D > C6 C6 90:9AB9 02 10<90> 08A8: 4D 89 00 32 DØ 02 A9 Ø888: 18 69 01 CO 14 RØ. OLA 99 ( BR ) 99 32<F4> 49 10 80 DISERS: ON 32 60 9D EB 29 F7(CD) 4C 28 60 **0**8C0: 69 24 24 27(90) **0**8C8: 29 23 2A 0F 119 33 57 MEDO: 2A 28 27 20 SE 24 4Bv 230 2A 3F 49(00> 24 30 63 0808: 2A 288 8: 30 63 30 57 30 30 22- BE > 08E8:30 A5 14 40 40 29 MA AACRA> AB Α9 02 14(60) 90 ØA. QUF 0: AS 24 08F8: 00 02 AB 98 89 Di 28 BD(4B> 29 94 28 80 A9 82 29(EE) 14(47) 8988:79 **R9** D2 70 24 09988+A5 ØA. AB 92 AB 88 89 01 28 8D< 05 / 0918:D0 29 29<04> 6919:AD 89 D2 28 8D AE 8928: BD 89 28 AD. 86 29 BD. C1 87 29 BD 28 BD(D9) 80 0928:28 29 28 BD 94 29<40> 0930:93 BD C2 0938 - BD 0.9 28 BD AB 29 BD EA< 97> AØ 08 Α9 00.C2> 0940:28 BD A1 29 34 99 00 99 0948:99 88 35 004BC> 0950 · 36 99 00 37 99 70 33 BB CB> 8C<4E> 29 0258:10 EE 94 8B 8C 2D Δ4 40 8968:83 29 80 8C 29 BC(7A) M9A9 . 9A 29 04 8D BC. 43 29 RE(29) 80 29 Δ4 BC\54> 0970: 9D 29 BE 46 29 8F 29 0978: AA 44 9C 49 BC(07> 0980:87 29 40 BB B9 33 38 DACES 8988: AB FØ 23 B9 6B 2B 99 22<28> 29 03(91) 0990134 89 CB A6 A9 FØ 0998:89 68 2B 99 88 35 B9 1Bx32> 09A8: 2A 66 00 FB 83 89 6B 2B(AZ) 09A8:99 24 ABKE22 99 36 B9 3F A6 37< 3C> 89 2B 99 00 0980: F0 03 68 8958:89 77 30 66 AC EQ 03 B9(38) 2B 99 89C8: 68 98 33 88 18 10 DD BE A6<AB> 09C8:88 A6 86 69 90 0900:87 FR AF 01 DØ A6 88 E8(12) 8E 03(8A) 0908: BE DØ 89 EB 02 A6 BA A6 E8 8E 07 DØ E8<24> 09EB: E8 8E 96 DØ E8 E9 BE 05<17> 09F8: 00 1C: 84 > E8 E8 BE 94 DØ 60 55(49) 09FB:3E 2A 69 7F 7E 55 36 0A00:55 14 10 3E TA 7B 7F(SC) 54 54 30 10(66) **DAGE: 7F** 55 54 16 55- 70> 0A10:3E 28 68 7F 7E 55 55 2E 34 7EKAC> 0A18:55 1,4 36 10 3E 6F 0A20: 7F 55 15 15 15 86 08 (8A) 7F (DA) 0A28:10 3E 10 10 3E 3E 7F 7F<13> 0A30: 3E 10 89 80 86 10 3E **BAJB: 3E** 3E 7F SE. LC. 88 88 20KAE> 1C<46> 3E<4F> 0A40:00 10 3E 80 7F 7F 3E 20 0A48:08 00 66 10 SE. 2A 8650: 7F 55 55 55 55 55 55 PRK 7F > @AS9:10 26 3E 3E 28 7F 55 55<8B> 0A60:55 55 14 99 60 10 3E 2A.8D> BA68: 2A 3E 7F 55 55 55 7F 41 **BRKSES** 24 3E 24 3E 55<F3> 0A70:10 55 @A78:55 55 55 20 04 A5 AE (51) 25 0A80:05 ΔF DØ 29 AD ØC DØ 25(09) FB A2 80 28<74> 00:00 B 85 20 88 0A90: AD 28 DØ 2D 89 DØ ØD ØAKE7) ØD 0A981 D0 12A BØ< 91> DØ 29 ØE 20 88 2000: F2 25 A2 24 2B A9<80> 0AA8: 01 DØ 88 DO(BA) BD 1E 60 AD ØA MARM: MD 09 DØ QD. DØ ØD BALBE > MARE: DO 29 ØF OD OC DØ 25 BØ< ØB > 49 48 PC. 25 2AC0: F2 AD DØ B8K6C> 77<4A> BACB: FR ØC **A5** 80 48 BD 0AD0:28 62 88 28 DF 26 AD @8<D3> 8408: D0 DØ ØA **0**0<73> ØD 99 ØD 90 DØ ØE 80 77 4A MAER: AS BPI 40 AA BD 2B A2<0B> DF 26 4A 98(09) 0AFR: 84 29 68 CB 85 A9 4A 04<61>

0000:A9 C8 85 AA 4A 90 04 A9<49> 0509:C8 85 AB A9 91 80 IE DØ< A9> A5 ΔØ 0.9 90<FA> 0B10:60 AØ. 16 0918:02 A9 ØF 85 A1 49 **ØF** 46(15) 9D 28 A5<4F> 0820:85 A2 AR) 17 20 8D(7F) 0928: A0 ØA ØA ØA ØA. 09 06 0B30: C4 02 49 RE ΒĎ CS 92 69K9F3 AD<63> 0838:4R AD C582 BD 17 DØ 0840:C6 82 8D 18 08 48 40 A9(12) 0848: 0C 80 17 DØ **A9** 88 BD 18<DE> 885-8: DB **A5** 1.4 29 RIF 49 DE SDC B50 AF 85 AE DØ CDF > 02 25 @858: C6 68 A2 24 85 AB FØ 02 D6(0F) 0860:08 10 @4<39> 8A:868 CA F7 68 98 02 88<1E> OR ZIE: RE DICH 51 50 50 50 51 10(75> 0578:00 (96) 10 3F SE 3E 3E 96<F6> 0880:00 88 20 188 96 96 64 64 ØF88: 44 32 49 88 AA 95 80<1A> ORSD: FR 10 EB 40 84 AR DO 31<82> 95(25) 2598: A9 95 79 27 19 AB BD 95 0BA0: B6 04(95) BD 7E 27 SR 49 26A8: 95 QDI PA FR 10 CA B3. 10KE2> 78 86 (DA) 09B0:15 AS A3 25 **A4** FØ 0BBB: 20 88 28 4C A5 29 20(ED) 69C0: 62 27 45 R3 30 24 20 1C(2E) 78 ASKEC) 66 CA 18 15 05C9: 2E 61 0900:A3 05 **A4** 96 20 28 (B2) 58 28 0808:4C 2B 20 83 27 AS(EA) CZ 20 A2KDB> ORFO: BI 04 10 2E 60 27 20 A2 AØ 88 **B9** 17<09> 0BE0: 08 10 @BF@:23 9D 98 32 CA 88 F6<98> 40<18> PINE REAS B1 25 83 10 CB 29 0000:2F **B7** 20 20 20 00(FA) 20 86 0C00:2C 4C 22 A9 80 48 20(A2) 10 E9<733 0C:0:53 25 20 83 27 68 38 00 44(BD) 20 9C18:81 DØ F3 68 0.0 DC20:44 90 20 AB (4A) 20 80 90 80 ØC28: AB 28 20 98 99 30 FC FC<DØ> FÇ 8C38: FC 20 80 00<43> 30 20 FF 0C38:00 88<44> 88 99 20 FF FF 99 BC48:88 20 99 20 FF EE FF BBC4A5 FF(51) 2C48: 20 20 20 20 FF FF FF 0C50:00 88 90 20 FF FF(59) OCSA: FF 0.0 20 20 FF SE EE FF(FA) 0C60:FF FF(AB) FF 200 (3)(3) FF FF FF 88 9C68: FF FF FF FF FF<000 3C 0C70:FF EE EF FF 86 7EKEC> 18 @C78: 7E 30 18 88 FF 81 81 BIK41> OC80:01 81 81 FF FF 28 88 99(81) 9C88: 89 98 00 88 90 00<90> 8C98:88 88 00 DO FF 20 00 **GBICARY** ØC98: ØØ 28 88 22 200< B(0) 20 28 6969 0CA0:00 99 90 99 10 10 10 10(A9) 0CA8: 10 00 99 88 99 88 60 15(00) @CB2:15 02 20 20 1.0 100 10 15<45> OCB8: 15 88 00 20 88 20 00 10(6B) 0CC0:10 10 10 10 10 10 10 10<00> QCC8: 10 1.18 1.00 10 (20) 200 22 15<84> 0CD0:15 10 173 16 15(70) 10 10 19 @CDB: 15 10 10 10 60 86 00 50(82) DOERL SO 20 88 200 10 10 10 50(D1) **OCEB:50** 69 55<7D> 80 88 99 88 307 OCF0:55 00 00 10 10 16 \$5(E8) OCF 8:55 69.01 90. 20 00 Oth 88 SACRAS 00000:50 10 10 18 10 10 10 50<7A> 0008:50 10 10 10 22 99 55<A6> @D1#:55 10 1.0 10 10 10 18 55<11> @D18:55 10 119 160 A9 68 AB 99(11) BD28: 08 38 99 90 39 99 3A< A5 > 90 @D28:99 88 3B CB DØ FI 60 AC<21> @D30: 22 D4 CØ 64 70 F9 CØ 7B(77) 0D30: B0 F5 A9 16 8D A9<60> 30 02 8D40:28 BD 31 92 AC D4(6F) 60 88 004B: CB 04 98 E9 CO 78 BØ F5(21) BD(65) 0050: A9 48 GD 30 A9 20 02 0058:31 02 60 20 AØ 26(44) 0060: A9 0.0 99 90 39 49 90 99(E3) FK 24(4A) **20068: BB** 39 88 10 20 68 ØD 70: 20 2D 80 A9(C2) 90 0078:04 85 85 94 20 25 Bd 20(32) @DB@: 20 62 27 B3<DC> 20 53 28 **2088:27** AD 88 D.S en. C9 ØF (C2) C9 17 0090:F8 38 FØ C9 00<25> ØE BD9B:FØ 18 C9 FØ. @7<2B> ØB. 86 C9 26 ODAC:FO 02 DØ **A5** B5 49 Ø1<1D> 80A8:85 95 AC 2D E6 **B**5 4C E6<03> 2D 0DB0:B5 4C AC C6 B5 C6 BS(BA)

0000:27 AD 00 D3 29 OF C9 OF(E6) 00C8: D0 F1 AD 18 De DØ **B**4 AD< 93> 0DD0:18 DØ FØ FB 46 **B**7 20 39<1A> **C6** B3(FF) ØDDB: 2D 20 62 27 **R1 C6** ØDE 0: 20 10 ZE 60 A5 **B**5 10 Ø3< 782 C9 70 E9(11) ODE8: 18 69 86 86 02 ADE Ø: Ø6 85 R5 AA RD 48 2F R5< 983 WDF8:B1 BD 4F ZE. **B**3 BD 5A(69) 85 0E 00: 2E 85 B6 BD 6C ZE. BD 31(35) 20 RE 28: 29 BD 72 **2E** 8D 32 BD(BD) ØE 10:54 2E R5 8A RD 68 **2£** 85(83) ØE18: 40 17 18(46) BD ZE 80 32 66 @E28:69 21 8D DE 22 20 05 28<67> 0E28:A2 A9 90 9D 00 90(93) ØË 32 A6<9B> ØF 38: 19 32 CA 1.0 F7 AB 88 A9 30 99 86(99) @E38: B1 86 **4B** CA 27(37) 0E40:32 CB DØ F7 A6 **B**3 AB 86 **BE48: A9** 4B CA 38 88 99 @8<87> ØF 50:32 DB F7 40 85 03 24 **Ø4<66>** 0E58:03 63 FF 65 FF 64 FF 83(FF) 00:00:00 CØ 20 CØ 88 CØ 03 04<75> **QE AR: D3** 64 83 84 FF SE 01 01(9F) 23 12 @E 70:03 10 10 12 14 14<E6> 0E78:6E 82 96 BE D2 20 20(77) ØE.80:20 20 20 20 98 40 44 4A<00> 0F 88: 80 40 40 40 40 (9)(7) 40 444355 0E 90: 4A 44 44 00 46 4A(83) 44 44 44 0E 98: 4A 98 60 88 88 90 88<83> 0F A0: 4A 00 40 20 40 00 40 ARC 185 0EA0:4A 00 20 90 46 48 89 4A<1A> 8E B8: 88 40 4A 40 48 99 44 88(12) DERRIAD NA AA 20 40 AA aΔ 00<33> 2EC8:4A 20 20 88 44 40 88 22<4F> 0EC8: 4A 90 44 90 4A 99 4A 00(89) OFTID: 44 90 20 88 20 46 48 48(99) 2ED8: 22 4A 86 88 44 20 44 **00**<24> 40<44> CEEO: 4A 66 44 88 4A 4A 44 ØFES: 4A 40 40 40 88 40 66 00<B2> DEFR: DR 4A (36) 40 40 40 40 PB(43) 0EF8: 4A 40 40 4A 40 20 98 4A<15> 44 0F00:00 86 88 00 44 00 4A<233 0F08:00 00 0401 an 40 00 00 AACRAS 0F 10:4A 20 40 22 46 22(BA) 22 22 0F18:4A 92 99 40 44 40 90 00(73) 88 0F20:4A AA 44 40 40 4A(70) 90 0F2B:00 00 40 202 44 88 20 46(23) 20<2D> 0F 30: 00 90 22 99 4A 20 40 0F38:4A AA 40 48 20 00 99 4A(05) 0F40:00 00 00 40 40 40 40 DOK BISS 0F 48: 4A AØ 80 66 40 13 89 78KE2> 0F50:2E FD 03 99 F9 89 BC(A1> FR 93 99 21 RE58: 2E 39 p9 ARLEES. @F 60: 2E FB 99 49 39 03 **B9** 84(61) 0F68:2E FØ 03 99 71 39 **B9** C8<20> DC(92) @F 70: 2F EB 0.3 99 90 39 89 @F 78: 2E FØ 99 3A 39 F8<A9> 03 89 0F 80: 2E FØ 03 99 61 3A B9 @4<15> REBB: 2E F0 03 99 89 3A **B9** 1B(DA) @F 98: 2F F9 93 99 **B**1 30 89 20(48) 0F98:2F FØ Ø3 99 109 3A 89 10(02) OF AD: AD 20 20 A1 99 A2 A2(6B) **BFAB: 48** 28 A1 2F 68 AP 20 BD(3C) 0F80:00 32 D9 92 90 DØKCDS 23 98 ØFR8: 07 CB 00 86 63 60KF8> CB<31>B5<74> RECO: BD RR 37 99 02 23 EB ØFC9:C0 FA F6 86 D8 60 B1 @FD@: B2 18 69 01 C9 10 02KDAX 20 00 **BFD8: A9** 19 95 B2 10 25 68<3A> OFE B: AT 98 A4 89 AB 82 **BB**<135 WFEB: A4 AA De 92 89 04 **A4** A9(45) 0FF0:00 02 89 02 85 BØ 60 A6<8D> OFF8:88 BD BD(57) 2F 300 9D 200 **B**2 1000:31 20 BD 61 D2 CA 30 182<68> 1000:86 88 AS AB AC BD 22(5E> 85 1010:02 FØ 82 A9 45 80 83 D2<10> 1018: A5 A3 69 D2<BD> 19 14 BD 24 1020:A5 A5 18 D2(36) 69 14 8p 106 1028: AS A3 F0 82 69 CA AD. MS(F3) 1838:02 **A5**5 A5 FØ 49 82 63 BD(E1) 1039:07 10(07) **D**2 60 28 SE 1949: TE 7E 7E 7E **7F** 7E 75 7F < 425 1048: 7F 3F 10 10 3E 78<D9> **7F** 7C 1050:70 70 78 7C 7F 10(22) 1058: 3F 7F 1E ØF 87 97 **PIF** 1E < 5D > 00 1060: 7F SE 10 22 63 63 63(19) 7F 10(53) 1068:77 77 7F 7F JE. 10 1070: 3F 7E **7**E 7E 77 77 63 63 (DE) 1078:63 22 80 00 00 00 1913 Ø8<D2>

Laenge 4219 Bytes

Listing zu »Twomaze« (Schluß)

53 25 20 83(62)

@D88:20 D8 2D 20

## AMPEL-Version 1.1

Mit AMPEL wird das Eingeben von reinen Atari-Maschinenprogrammen zum Kinderspiel.

peziell ber Maschinenprogrammen kommt es darauf an, daß jedes Bytes korrekt eingegeben wird. Schon ein einziger falscher Wert führt meist dazu, daß das eingegebene Programm nicht läuft. Dies hat dann oft eine stundenlange Fehlersuche zur Folge. Deshalb haben wir für Sie »AMPEL« (Atari-Maschinen-Programm-Eingabe-Listing) entwickeln lassen. Und damit das Eintippen für Sie möglichst einfach und reibungslos verläuft, wählen wir eine spezielle Darstellungsform für Maschinenprogramme. Beachten Sie bitte, daß Sie zukünftig unbedingt AMPEL-Version 1.1 verwenden müssen, um Atari-Maschinenprogramme einzugeben. Von Basic aus lassen sich solche Listings nicht eingeben. Verwahren Sie also das Programm zu AMPEL sorgfältig und legen Sie eine zusätzliche Kopie an einen sicheren Platz.

Geben Sie zunächst das nachfolgende Listing ein. Verwenden Sie dazu den Atari-Prüfsummer. Bevor Sie AMPEL mit RUN starten, sollten Sie das eingegebene Basic-Programm auf Diskette oder Kassette speichern. Hat sich nämlich ein Fehler im Maschinenspracheteil eingeschlichen, kann Ihr Computer abstürzen. In dem Fall müßte AMPEL erneut eingegeben werden.

#### Grünes und rotes Licht mit AMPEL

Nachdem Sie das Programm mit RUN gestartet haben, werden zuerst die DATA-Werke eingePOKEt. Dieser Vorgang nimmt einige Sekunden in Anspruch. Danach müssen Sie die Länge des einzugebenden Maschinenprogramms eingeben. Diesen Wert entnehmen Sie bitte am Ende des abgedruckten Maschinenprogramms. Anschließend geben Sie den Programmnamen an, unter dem Ihr Programm gespeichert wird. Auf dem Bildschirm erscheint jetzt »0000:«, die Aufforderung, mit der Eingabe der ersten Zeile des AMPEL-Listings zu beginnen.

Betrachten wir eine AMPEL-Listing-Zeile aus der Nähe:

0000:00 02 00 07 06 07 20 BD (F3A)

Die ersten vier Zahlen (hier 0000) stellen sozusagen die Zeilennummer dar. Diese Zahlen sowie den darauffolgenden Doppelpunkt brauchen Sie nicht einzugeben. Danach folgen acht zweistellige Hexadezimalzahlen, die Sie dem Listing entnehmen. Die Eingabe erfolgt ohne Betätigung der RETURNTaste. Der Cursor springt selbständig von einer Position zur nächsten. Leerzeichen sowie die beiden Zeichen »(« und »)« fügt AMPEL automatisch ein. Ist die Prüfsumme korrekt – sie muß auch von Ihnen eingegeben werden – springt der Cursor in die nächste Zeile. Falls nicht, ertönt ein akustisches Signal und alle acht Werte, inklusive der Prüfsumme, müssen nochmals eingegeben werden.

Außer den Hex-Tasten (0 bis 9 und A bis F), wird nur noch DELETE BACKSPACE zum Korrigieren des letzten Zeichens und CONTROL Sabgefragt. Bitte beachten Sie den Kasten

»Achtung Änderung«!

#### Zwischenspeichern

Da Sie ein umfangreiches Maschinenprogramm sicher nicht an einem Tag eintippen möchten, können Sie zu jedem Zeitpunkt mit CONTROL S zwischenspeichern. Dazu betätigen Sie einfach CONTROL S und der Code wird unter dem zuvor eingegebenen Programmnamen gespeichert. Weiterhin wird automatisch die jeweils zuletzt gespeicherte Version in »BACKUPOBJ« umbenannt. Zuvor wird natürlich eine ältere »BACKUPOBJ«-Datei gelöscht. Danach können Sie den Computer ausschalten oder AMPEL mit RESET verlassen.

Wenn Sie das nächstemal AMPEL starten, muß wieder die Programmlänge und der entsprechende Name, mit dem Sie Ihre letzte Version gespeichert haben, eingegeben werden. Existiert bereits eine Datei mit dem entsprechenden Namen, wird die Datei automatisch geladen. Weiterhin wird noch die Zeilennummer auf dem Bildschirm angezeigt, ab der Sie mit der Eingabe der Hex-Werte fortfahren müssen.

Speichern Sie lieber öfter mit CONTROL S ab, um auch gegen eventuelle Stromausfälle gewappnet zu sein. Weiterhin empfiehlt es sich aus Sicherheitsgründen, zwei Disketten für Daten vorzusehen. Wenn Sie das komplette Programm eingegeben haben, speichert der Computer nach Eingabe des letzten Bytes das Programm auf Diskette ab und meldet sich danach mit der READY-Meldung.

Wenn Sie Ihr Maschinenprogramm anschließend laden möchten, erfolgt dies vom DOS-Menü aus. Wählen Sie hier die L-Funktion und geben Sie anschließend den entsprechenden Programmnamen ein.

#### Besondere Hinweise für Kassettenrecorder-Besitzer

Damit AMPEL auch einwandfrei mit einem Kassettenrecorder funktioniert, müssen folgende Programmzeilen geändert werden (beachten Sie bitte auch die Änderungshinweise »Achtung Änderung«):

290 F\$="C:" :AD=0
300 ? "Altes F1le laden J/N ";
310 INPUT FR\$
320 IF FR\$( ) "J" THEN 510
420 TRAP 510:OPEN #1,4,128,F\$
590 TRAP 660:CLOSE#1:OPEN #1,8,128,F\$

Die Zeilen 330 bis 360 entfallen, ebenso die Zeilen 450, 600 und 610

Speichern Sie anschließend die geänderte AMPEL-Version mit CSAVE ab. Somit haben Sie das benötigte Eingabepro-

PROG	RAMM-STECKBRIEF
Programmame	AMPEL
Programmtyp	Eingabehilfe
Programmiersprache	Atari-Basic
Programmlänge	3965 Byte
für Computer	alle
zusätzliche Hardware	Diskettenlaufwerk oder Kassettenrecorder
Eingabehilfe	Prüfsummer
Bemerkung	komfortable Eingabehltfe für reine Maschinenprogramme. Besitzer eines Kassettenrecorders müssen ein zusätzliches Programm eingeben
Leserservice	Diskette (AMPELBAS)

#### EINGASESHI-FEN

gramm bereits vorliegen. Damit aber Binary-DOS-Files (Maschinenprogramme) von Kassette aus geladen und gestartet werden können, müssen Sie ein zusätzliches Hilfsprogramm eingeben. Dazu muß unbedingt AMPEL verwendet werden.

Beginnen Sie also, indem Sie RUN eingeben. Als Programmnamen geben Sie »C:« (bitte den Doppelpunkt nicht vergessen) und auf die Frage nach der Programmlänge 198 ein (die Programmlänge finden Sie stets am Ende eines AMPEL-Listings). Geben Sie Wert für Wert ein, bis sich das »Kassetten-Hilfsprogramm« vollständig im Speicher befindet. Der Computer meldet sich nach dem letzten Byte mit einem Brummton, der Sie auffordert, die RECORD- und PLAY-Tasten des Recorders zu betätigen. Nach Drücken der RETURN-Taste wird das kurze Hilfsprogramm auf Kassette gespeichert. Speichern Sie es von vornherein am Anfang von verschiedenen Kassetten ab, da es Grundvoraussetzung für das Laden von Maschinenprogrammen von Kassette ist. Notieren Sie sich unbedingt den Zählerstand, um es nicht versehentlich zu löschen.

#### Sicherheit über alles

Mit CONTROL S kann zu jedem Zeitpunkt gespeichert werden. Daraufhin erklingt ein doppelter Signalton. Betätigen Sie dann noch die RECORD- und PLAY-Tasten des Recorders und anschließend die RETURN-Taste. Der Speichervorgang beginnt. Wenn Sie dann mit einem anderen Programm arbeiten möchten, gelangen Sie mit RESET zurück ins Basic.

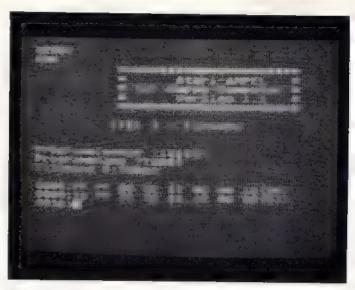
Wenn Sie zu einem späteren Zeitpunkt mit der Arbeit fortfahren möchten, laden Sie wieder zuerst AMPEL, starten es mit RUN, geben die Programmlänge ein und beantworten die Frage »Altes File laden J/N« mit J. Daraufhin meldet sich der Computer wieder mit einem Brummton. Spulen Sie das Band an den Anfang zurück, betätigen Sie dann noch die PLAY-Taste und RETURN. Jetzt wird der zuletzt eingegebene Programmteil geladen. Anschließend meldet sich der Atari-Computer wieder mit der Zeilennummer, ab der Sie mit der Eingabe fortfahren müssen.

Um später das eingegebene Programm zu starten, muß stets zuerst das »Kassetten-Hilfsprogramm« geladen werden. Nach dem Ladevorgang meldet sich der Computer mit dem Brummton. Anschließend legen Sie bitte Ihre Kassette mit Ihrem Maschinenprogramm in den Recorder ein, drücken die PLAY-Taste des Recorders und während Sie die START-und OPTION-Tasten des Computers drücken, schalten Sie die Stromversorgung ein. Nach dem Brummton nur noch RETURN drücken und Ihr Programm wird geladen und anschließend automatisch gestartet.

#### Programme schnell geladen

Speziell bei Kassettenrecordern kann es Probleme beim Laden von Maschinenprogrammen geben. Deshalb sollten Sie stets mehrere Sicherheitskopien von Ihren Programmen machen. Hinzu kommt, daß das Laden von Kassette sehr viel Zeit beanspruchen kann, da die Übertragungsgeschwindigkelt sehr langsam ist. Besitzer einer Diskettenstation haben es hier viel besser. Also nicht vergessen: Wenn Sie als Besitzer eines Kassettenrecorders Maschinensprache-Programme laden wollen, müssen Sie stets zuerst das Kassetten-Hilfsprogramm laden. Sollten Sie mit AMPEL auf Schwierigkeiten stoßen, wenden Sie sich an die Redaktion.

(Frank Ostrowski/wb)



So sieht der Bildschirm bei der Eingabe von Maschlnenprogrammen aus. Zwischenspelchern kann man jederzelt mit CONTROL S.

#### Achtung Änderung!

Wenn Sie AMPEL aus Happy-Computer, Ausgabe 12/85 bereits eingetippt haben, müssen Sie eine kleine Korrektur vornehmen, damit die DELETE/BACKSPACE-Funktion korrekt ausgeführt wird. In den Zeilen 1230, 1660, 1750 hat sich jeweils ein kleiner Fehler eingeschlichen. So sollten die Zeilen aussehen:

1230 IF A=8135 THEN LIST 1650,1750 1660 DATA 102, 221, 102, 222, 102, 223 1750 DATA 8135

Speichem Sie die neue Version auf Ihrer Masterdiskette. Zur Eingabe von reinen Maschinenprogrammen sollten Sie die AMPEL-Version 1.1 verwenden.

> 8888:88 82 88 87 86 87 20 9D<3A> 9998:07 A2 18 A9 83 9D 42 93<D6> 8818:49 C3 9D 44 83 A9 87 9D<D3> 9828:88 9D 4B 83 28 56 E4 38<73> 0028:6D A9 97 A0 87 8D E0 02<22> 9938:80 E1 92 28 98 97 C9 FF(6A) 9838:D9 5C CB 08 59 A9 97 A8<75> 8848:87 BD E2 82 BC E3 82 28<97> 8848198 87 C9 FF D8 84 C8 FF<75> 0050:FB EB 9D 44 83 98 9D 45(EF) 9B 07 39 FD 44 03<05> 8628:63 29 8068:90 48 83 98 FD 45 83 9D<Re> 0068:49 03 FE 48 03 D0 03 FE(DC) 0070:49 03 28 56 E4 39 1F 20(FC) 8079:98 87 AD 53 83 C9 83 D8(F2) 9080:BC A2 19 A9 9C 9D 42 83(32) 0088:28 56 E4 38 89 28 BD 07<F8> 0070:18 6C E0 02 68 68 38 60<85> 9098:6C E2 82 A9 07 A2 10 9D(8D) 00A0:42 83 A9 80 9D 48 83 90<68> 00A8:49 03 20 56 E4 38 D2 48<42> 00B8:A9 00 9D 48 83 20 56 E4<64> 9288:30 DA A8 48 48 A9 3C 8D<7D> 00C0:02 03 60 43 3A 98 00 Laenge 198 Bytes

Listing »Kassetten-Hilfsprogramm« (Bitte mit AMPEL eingeben)

180 ? ,"_====================================	<hi></hi>	1240 IF A=7184 THEN LIST 1760,1860	<xk></xk>
110 ? ""=*====Atari=AMPELasa*"	< XD>	1250 IF A=2222 THEN LIST 1870,1910	<0U>
115 ? "=#=(c)=Happy-Computer=#"	<jt></jt>	1300 DATA 173,37,228,72,173,36	<yx></yx>
117 ? , "a*aaaaaVersional.laaa*"	<oe></oe>	1310 DATA_228,72,96,72,74,74	<mi></mi>
128 7 "-***********************************	<hh></hh>	1329 DATA_74,74,32,29,6,104	<za></za>
130 ? 140 ? ."CTRL_S_=_Sichern"	<2n>	1330 DATA_41,15,9,48,201,58	<30>
150 ?	(ZY)	1340 DATA_144,2,105,6,168,173	<oh></oh>
168 ? "Einen_Moment_Geduld_bittm.";	<b>(14)</b>	1350 DATA_71,3,72,173,70,3	<dm></dm>
190 POKE 16,64:POKE 53774.64	<cl></cl>	1360 DATA_72,152,96,50,31,30	<cg></cg>
230 DIM F\$(15),FB\$(15),FR\$(27)	<wk></wk>	1370 DATA 26,24,29,27,51,53	<il></il>
240 DIM ML*(261),CIO*(83)	<hh></hh>	1380 DATA_40,63,21,18,58,42	<kv></kv>
250 GOSUB 1880	<3T>	1390 DATA_56,190,52	(PY)
260 7 CHR\$(156)	<dg></dg>	1430 DATA_169,0,133,213,104,201 1440 DATA_2,240,16,170,240,5	<hx></hx>
278 7 "Programmlamnges"::INPUT L	⟨EH⟩	· · · · · · · · · · · · · · · · · · ·	<yr></yr>
280 DIM A*(L+8)	<6W>	1450 DATA_104,184,202,208,251,169 1460 DATA_3,44,169,1,133,212	<rj> <bu></bu></rj>
290 ? "Programmame.";:INPUT F#	<ir></ir>	147@ DATA_96,104,133,215,104,133	<yk></yk>
300 IF F\$="" THEN 290	<ma></ma>	1480 DATA_214,104,133,217,104,133	<00>
318 F=8:FOR I=1 TO LEN(F#):F=F+(F#(I,I)=	11417	1498 DATA_216,24,181,214,133,214	(UF)
"1") 1NEXT I	<ir></ir>	1500 DATA_165,217,101,215,133,215	<iu></iu>
320 IF F=0 THEN FR\$="D:":FR\$(3)=F\$:F\$=FR	12117	1510 DATA_169,155,32,28,6,165	<8R>
*	(UE)	1520 DATA_217,32,9,6,165,216	<lg></lg>
338 ? "Filename_":F#	<ta></ta>	1538 DATA_7931	<ms></ms>
340 FB*="D:BACKUP.OBJ"	<nh></nh>	1540 DATA_32,9,6,169,58,32	<dx></dx>
350 FR#=F# FR#(LEN(FR#)+1)="."	<xt></xt>	1550 DATA_28,6,169,0,133,218	<j@></j@>
360 FR\$(LEN(FR\$)+1)=FB\$(3)	<10>	1568 DATA-173,252,2,41,191,162	(NB)
420 AD=0:TRAP 510:OPEN #1.4.0.F\$	<0M>	1570 DATA_17,221,39,6,240,5	<dp></dp>
430 E=USR(ADR(CIO\$),1,7,ADR(A\$),L)	<zh></zh>	1500 DATA_202,16,240,40,239,134	<8Q>
440 IF E=1 THEN ? "Schon_fertig":END	<in></in>	1590 DATA_219,141,252,2,32,0	<um></um>
450 IF E<>136 THEN ? "Disk-Read-Error.";		1600 DATA_6.41.127.32.20.6	<gb></gb>
ELEND	<ql></ql>	1618 DATA-165,219,281,16,248,168	<1B>
460 AD=PEEK(856)+256*PEEK(857)	<qt></qt>	1620 DATA_144,40,165,218,240,205	(EFX
470 AD=8*INT(AD/8):TRAP 65535	< <b>VQ</b> >	1630 DATA_74,176,5,169,126,32	<rs></rs>
510 CD=USR(ADR(ML*),ADR(A*),AD)	<xm></xm>	1648 DATAL6585	<ph></ph>
520 ON CD GOSUB 590,690,710	<pw></pw>	1650 DATA_28,6,160,4,70,220	<yy></yy>
530 AD=AD+8+(CD=0)	<8V>	1660 DATA_102,221,102,222,102,223	<ve></ve>
540 IF ADC=L THEN 510 (1407)	<wk></wk>	1670 DATA_102,224,102,225,102,226	<dy></dy>
558 AD=L: GOSUB 590 \$53774, 192	<ni></ni>	1680 DATA_102,227,102,228,136,208	<qr></qr>
	(KDZ>)?	1670 DATA_235,198,218,184,80,178	<ba></ba>
590 TRAP 600:CLOSE #1:XIO 33,#1,0,0,FB\$	<md></md>	1700 DATA_10,10,10,10,160,4	<l0></l0>
600 TRAP 610:CLDSE #1:XID 32,#1,0,0,FR\$	F	1716 DATA_10,38,228,38,227,38	<uu></uu>
610 TRAP 660: CLOSE #1: OPEN #1,0,0,F\$	<md></md>	1720 DATA_226,38,225,38,224,38	<yp></yp>
620 E=USR(ADR(CIO\$),1,11,ADR(A\$),AD)	<lc></lc>	1738 DATA_223,38,222,38,221,38	<ru></ru>
430 CLOSE #1	<li></li>	1740 DATA_220,136,208,234,230,218	<06>
640 IF E<>1 THEN 7 "Disk-Errora"; E 650 TRAP 65535: RETURN	<zn> <fb></fb></zn>	1750 DATA_8135	<lz></lz>
650 TRAP 65555: RETURN 650 CLOSE #1	(LD)	1768 DATA_165,218,201,18,248,19	<wo></wo>
476 D. NO L. C N. DEEK (1951 - DET) DAI	<xxx></xxx>	1778 DATA_281,16,288,4,169,68	<mk></mk>
670 ? "Disk-Error.";PEEK(195):RETURN 690 ? CHR\$(253):? "Prumfsummenfehler"	<pd></pd>	1780 DATA_208,3,74,176,206,169	<e8></e8>
700 RETURN	<mr></mr>	1790 DATA_32,32,28,6,184,88	<3Z>
1 == 1 = 1 = 1 = 1	<cm></cm>	1806 DATA_198,165,217,18,181,216	(ZP)
718 ? "Fehler!!!!";:LIST 510:END	<wd></wd>	1818 DATA_10,181,220,10,181,221 1828 DATA_10,101,222,18,101,223	<u0></u0>
1010 FOR I=1536 TO 1592	<nu></nu>	1838 DATA_18_181_224_18_181_225	<yp></yp>
1020 READ A:Q=Q+A:POKE I,A	<kx></kx>	1848 DATA_10,101,226,18,101,227	<cq></cq>
1938 NEXT I	⟨EY⟩	1850 DATA_197,229,288,18,169,62	<nj></nj>
1040 IF Q<>4196 THEN ? "Data_Error":LIST		1868 DATA_7184	<ns></ns>
1300,1390; END	<jv></jv>	1878 DATA_32,28,6,168,7,185	<ll></ll>
1000 CI-N	<xd></xd>	1880 DATA_220,0,145,214,136,16	<kj></kj>
1868 FOR I=1 TO 261	<rz></rz>	1898 DATA_248,169,8,44,169,2	<nu></nu>
1070 READ A	<10>	1900 DATA_133,212,96	(CA>
1080 IF A>255 THEN SCSUB 1170	(EL)	1918 DATA_2222	<hh></hh>
1898 ML*(I)=CHR*(A):Q=Q+A	<hu></hu>	1958 DATA_184,281,4,288,66,184	<ke></ke>
1100 NEXT I	<er></er>	1968 DATA_104,170,184,184,133,212	<ax></ax>
1118 READ A: IF QC>A THEN 1288	(FZ)	1976 DATA_104,133,214,164,133,213	<dh></dh>
1120 Q=0	<hh></hh>	1988 DATA_104,133,216,104,133,215	<ho></ho>
1130 FOR I=1 TO 83	<tl></tl>	1998 DATA_168,134,138,281,8,176	<ax></ax>
1140 READ A: 0=0+A:CIO\$(I)=CHR\$(A)	<bl></bl>	2000 DATA_33,10,10,10,10,170	<lb></lb>
1158 NEXT I	<f6></f6>	2010 DATA_165,212,157,66,3,165	<vr></vr>
1168 IF GK>10894 THEN ? "Data_Error_";:L		2020 DATA_213,157,68,3,165,214	<to></to>
IST 1950,2000: END	<mp></mp>	2830 DATA_157,69,3,165,215,157	<ad></ad>
1170 RETURN	<pq></pq>	2040 DATA_72,3,165,216,157,73	<se></se>
1190 IF Q=A THEN READ A:Q=0:RETURN	<va></va>	2056 DATA_3,32,86,228,132,212	<hm></hm>
1200 ? "Data_Error"	<60>	2060 DATA_169,0,133,213,96,160	<0U>
1210 IF A=7931 THEN LIST 1430,1530	<iu></iu>	2070 DATA_10,170,240,244,104,104	<5B>
1220 IF A=6585 THEN LIST 1540,1640	<px></px>	2080 DATA_202,208,251,240,237	<25>
1230 IF A=8135 THEN LIST 1650,1750	<rd></rd>		

Listing zu »AMPEL«



# Prüfsummer paßt auf

Unser Prüfsummer für alle Atarl-Computer ist besonders komfortabel. Mit Hilfe dieses Programms lassen sich Listings auf Anhieb korrekt eingeben.

ag der Atari-Prüfsummer auf den ersten Blick etwas umfangreich erscheinen, der Aufwand lohnt sich. So legt der Prüfsummer zum Beispiel eine zusätzliche Statuszeile am oberen Bildschirmrand an. Diese Zeile ist von Basic aus nicht zu erreichen, also auch nicht zu löschen. Welterhin können Sie den Prüfsummer auch für die Eingabe von Turbo-Basic XL-Programmen verwenden. Beachten Sie hierzu die besonderen Hinweise!

In dem abgedruckten Listing finden Sie gewisse Zeichen vor, die der Atari nicht kennt. Es handelt sich dabei einerseits um Dreiecke und andererseits um geschweifte Klammern. Die Dreiecke stellen grundsätzlich Leerstellen dar; Texte, die zwischen geschweiften Klammern stehen, respräsentieren Atari-spezifische Grafik- und Sonderzeichen. Ein Beispiel: »ESC CTL=« entspricht dem ASCII-Zeichen 29. Um dieses Zeichen einzugeben, betätigen Sie zunächst die ESC-Taste und dann gleichzeitig die CONTROL- und die »= «-Taste. Daraufhin erscheint das gewünschte Zeichen auf dem Bildschirm. Eine komplette Aufstellung aller Grafik- und Sonderzeichen finden Sie in der Tabelle.

Weiterhin gibt es unterstrichene Zeichen, die eine inverse Darstellung bedeuten. Bevor Sie ein solches Zeichen eingeben, müssen Sie unbedingt die Atari-Taste betätigen. Bild 1 zeigt eine Programmzelle in der üblichen Darstellungsweise und zum Vergleich dazu eine in konvertierter Form.

Geben Sie also zunächst das Prüfsummenprogramm von Basic – oder Turbo-Basic XL – aus ein. Zwar werden sämtliche DATA-Werte nach dem Start mit RUN überprüft, aber »nobody is perfect«.

Kontrollieren Sie also das Prüfsummenprogramm vor dem Start noch einmal. Speichern Sie es dann auf einem Datenträger. Jetzt können Sie das Programm beruhigt mit RUN starten. Aus den DATA-Werten wird nun ein Maschinen-Programm erzeugt. Sollte sich beim Eintippen des Programms ein falscher DATA-Wert eingeschlichen haben, so wird die entsprechende Zeilennummer auf dem Bildschirm angezeigt.

Nach der Korrektur starten Sie das Programm erneut mit RUN. Wurde das Programm fehlerlos eingegeben, stehen folgende Funktionen zur Auswahl:

1. Das Prüfsummen-Programm als Maschinensprach-File auf Diskette schreiben. Nach Betätigung der Taste 1 fragt das Programm nach dem Namen für das Prüfsummenprogramm. Tippen Sie also beispielswelse »PRUEFER.OBJ« ein, so wird nach Druck auf dle RETURN-Taste das Prüfsummenprogramm auf Diskette gespeichert. Der Computer schließt den Vorgang mit der READY-Meldung ab.

Um das Prüfcode-Programm verwenden zu können, löschen Sie den Programmspeicher zunächst mit NEW. Vom DOS-Menü aus (einfach DOS eintippen und die RETURN-Taste betätigen) wählen Sie die Option L und geben den Namen des vorher erzeugten Maschinenprogramms ein. Jetzt wird das Prüfsummenprogramm in den RAM-Speicher geladen. Mit B gelangen Sie wieder zurück ins Basic.

Aktiviert wird jetzt das Prüfsummenprogramm mit >?USR(1536) . Am oberen Bildschirmrand erhalten Sie daraufhin eine zusätzliche Zeile mit dem Schriftzug 4850 POSITION 11,2:7 "L POSITION 4,5:7 "FILENAME SIZE":POSITI ON 2.7

4850 POSITION 11,2:? "(CTL 2)(CTL R)(CTL R)(

Bild 1. Oben eine Programmzeile, wie sie auf dem Bildschirm erscheint. Darunter dieselbe Zeile in der konvertierten Form.



Bild 2. So sollte der Bildschirm nach dem Aktivieren des Prüfsummers aussehen.

»Prüfcode: > < (c) by W. Kress/wb«. Wie die Aufteilung des Bildschirms unter Verwendung des Prüfsummenprogramms aussleht, zeigt Bild 2.

 Das Maschnenprogramm wird als Boot-File auf Kassette geschrieben. Das spätere Laden oder Booten geht folgendermaßen vor sich:

 bel ausgeschaltetem Computer die START-Taste drücken und Computer einschalten,

 nach Ertönen des Summtons die START-Taste loslassen
 nacheinander dle PLAY-Taste des Kassettenrecorders und die RETURN-Taste betätigen.

das Prüfsummenprogramm wird geladen und automatisch gestartet

3. Programm beenden. Das Maschinenprogramm wird nicht gespeichert.

PROGRAMM-STECKBRIEF						
Programmname	Prüfsummer					
Programmtyp	Eingabehilfe					
Programmiersprache	Atari-Basic und Maschinensprache					
Programmlänge	4075 Byte					
für Computer	alle					
zusätzliche Hardware	Diskettenstation oder Kassettenrecorder					
Eingabehilfe	kelne					
Bemerkung	Eingabehilfe für Atari-Basic und Turbo-Basic-XL-Listings.					
Leserservice	Diskette (PRUEFSUM BAS)					

Das Prüfsummenprogramm wird in der Page 6 abgelegt, einem vor NEW geschützten Speicherbereich. Abschalten kann man es durch Drücken von SYSTEM RESET, ein Neustart erfolgt mit dem Aufruf: \*?USR(1536)\*.

Sobald das Prüfsummenprogramm aktiviert ist, können beim Zugriff auf ein Speichermedium Bildstörungen auftreten. Um dies zu vermeiden, brauchen Sie nach dem Laden oder nach dem Aufruf mit »?USR(1536)« nur »POKE 39998,30« einzugeben.

Wurde das Programm von Kassette gebootet, wird es jedesmal nach Betätigung der SYSTEM RESET-Taste erneut gestartet. Gibt man jedoch vor der Betätigung der SYSTEM RESET-Taste »POKE 9,0« ein, kann man diesen Effekt umgehen.

Bevor Sie nun ein eingetipptes Basic- oder Turbo-Basic XL-Programm mit RUN starten, sollten Sie es sicherheitshalber noch einmal speichern. Weiterhin sollte das Prüfsummen-Programm mit SYSTEM RESET desaktiviert werden, da auch manche Basic-Programme die Page 6 für Maschinenunterprogramme verwenden. Es könnte sonst zu Überschneidungen und zum Systemabsturz kommen.

Der Prüfcode besteht jeweils aus zwei Zeichen. Er setzt sich aus den Buchstaben von A bis einschließlich Z zusammen. Geben Sie nun probeweise eine Programmzeile ein und schließen Sie mit der RETURN-Taste ab. Der zugehörige Prüfcode erscheint dann in der obersten Zeile. Sollte der Code nicht mit dem im abgedruckten Listing

übereinstimmen, kann die Programmzeile sofort verbessert werden. Die jeweilige Programmzeile braucht dazu nicht erneut eingegeben werden, da sämtliche Editierfunktionen des Atari-Computers zur Verfügung stehen.

Ein wichtiger Hinweis:

Für die erste Programmzeile kann eventuell ein falscher Prüfcode errechnet werden. Sollte die Prüfsumme also nicht mit der im Listing abgedruckten übereinstimmen, fahren Sie mit dem Cursor nochmals auf die erste Zeile. Sobald Sie die RETURN-Taste betätigt haben, erscheint der richtige Code auf dem Bildschirm. Stimmt der ausgegebene Wert dann immer noch nicht, überprüfen Sie nochmals jedes einzelne Zeichen der von ihnen eingegebenen Programmzeile.

Bei der Eingabe von Programmen sind die üblichen Abkürzungen für Basic-Befehle zulässig. Sie können also zum Beispiel G. für GOTO, SE für SETCOLOR und so weiter verwenden. Eine Ausnahme gilt jedoch bei der Abkürzung für den PRINT-Befehl, den man in der Regel mit einem Fragezeichen abkürzen kann. Verwenden Sie das Fragezeichen bitte nur dann, wenn es auch im Listing verwendet wird.

Möchten Sie nachträglich ein Programm überprüfen, so laden Sie zuerst das Prüfsummenprogramm und starten es mit »?USR(1536)«. Anschließend laden Sie Ihr Programm und listen einen Teil auf dem Bildschirm. Dann bewegen Sie den Cursor an irgendeine Stelle einer Programmzeile. Mit

Code	ATASCII Zeichen	zu belättgende Taste(n)	Dez:mal Code	ATASCII Zeichen	2. betängende Taste(n)	Dezimal Code	ATASCII	zu beiängende Tastein)	Dezimal- Code	ATASCII Zeichen	z. betatigende Tas'ë(n)
A	В	С	Ā	B	С	A	В	С	A	В	С
0	•	CTL	19	+	CTLS	128	V	(Invers) CTL	147		(Invers) CTL S
ı	D	CTL A	20		CTL T	129	F	(Invers) CTL A	148		(Invers CTLT
2		CTL B	21		CTL U	130		(Invers) CTL B	149		(Invers) CTL U
3		CTLC	22		CTL V	131	4	(Invers) CTL C	150		(Invers) CTL V
4	1	CTL D	23	7	CTL W	132		(Invers) CTL D	151	-	(Invers) CTL W
5		CTL E	24		CTL X	133	5	(Invers) CTL E	192		(Invers) CTL X
6		CTL F	25		CTL Y	134		(Invers) CTL F	.53 j		(Invers) CTL Y
1		CTL G	26	•	CTL Z	135		(Invers) CTL G	154	L	(Invers) CTL Z
8		CTL H	27	Ę	ESC ESC	136		(Invers) CTL H	.56	1	ESC SHIFT DEL
9		CTL I	28	1	ESC CTL	137		(Invers) CTL I	157	Ŧ	ESC SHIFT >
0		CTL J	29	•	ESC CTL =	136	N	(Invers) CTL J	158	÷	ESC CTL TAB
.		CTL K	30	+	ESC CTL +	139		(Invers) CTL K	159	<b>→</b>	ESC SHIFT TAB
2		CTL L	.31	•	ESC CTL *	140		(Invers) CTL L	224		(Invers) CTL
3		CTL M	96	•	CTL	141		(Invers) CTL M	251	•	(Invers) CTL
4		CTL N	123	•	CTL,	ı42		(Invers) CTL N	252		(Invers) SHIFT
5		CTL O	.24	Ш	SHIFT	143		(Invers) CTL C	253	5	ESC CTL 2
5		CTL P	125	<b>F</b>	ESC CTL «	144	*	(Invers) CTL P	254		ESC CTL DEL
7		CTLO	126	1	ESC DEL	145	G	(Invers) CTL Q	255		ESC CTL >
9		CTL R	127		ESC TAB	146		(Invers) CTL R	Tabel	le de	Atari-Grafikzeichen

#### engari-sugean

Betätigung der RETURN-Taste wird der Prüfcode in der Statuszeile dargestellt. Stimmt der Prüfcode mit dem im beigefügten Listing überein, betätigen Sie wieder die RETURN-Taste, und der Prüfcode der nächsten Zeile wird errechnet und angezeigt. Auf diese Art und Weise können Sie ein komplettes Programm oder auch nur einzelne Zeilen »durchchecken«.

Sollte der Code nicht mit dem des Listings übereinstimmen, überprüfen Sie Ihre Programmzeile. Denn bei allen abgetippten Listings gilt: Die Programme müssen so abgetippt werden, wie sie abgedruckt sind. Stimmt die Anzahl der Leerzeichen? Haben Sie eventuell Zeichen vertauscht? Oder haben Sie ein »?« anstelle eines PRINT-Befehls eingegeben? Haben Sie den Fehler gefunden? Dann weiterhin viel Erfolg mit Ihren Atari-Programmen.

#### Besondere Hinweise zum Prüfsummenprogramm

Wenn ihnen die Redaktion unseren Atari-Prüfsummer bereits zugeschickt hat, müssen Sie eine geringfügige Änderung am Basic-Programm vornehmen. In der alten Version funktioniert der Prüfsummer zwar bei der Eingabe von normalen Basic-Programmen, nicht aber bei Turbo-Basic XL-Programmen. Sie müssen aber lediglich am Ende von Zeile 630 ein Semikolon »;« anfügen. Sie sollte dann wie folgt aussehen:

630 IF D THEN ? #1;D\$(1,500);

Starten Sie das Prüfsummenprogramm anschließend mit RUN und speichern Sie es wieder auf Ihrem Massenspeicher. Auf der Leserservice-Diskette befindet sich natürlich die neue, hier abgedruckte Prüfsummer-Version.

Wenn Sie ein Turbo-Basic XL-Programm eingeben möchten, müssen Sie das Prüfsummenprogramm mit einem Hilfsprogramm starten. Weiterhin empfiehlt es sich, eine separate Diskette, sozusagen als Eingabediskette zu verwenden. Formatieren Sie also eine Diskette. Verwenden Sie aber bitte nur DOS 2.0 oder 2.5! Speichern Sie anschließend die DOS-Files und Turbo-Basic XL mit dem Namen »AUTORUN.SYS« auf dieser Diskette. Anschließend schalten Sie den Computer aus und booten die soeben angelegte Diskette. Geben Sie anschließend folgende Programmzeile ein:

1 BLOAD "D:PRUEFER.OBJ":POKE \$BC3E,\$1E:\*L-:NEW

Speichern Sie diese Programmzeile dann mit »SAVE" D:AUTORUN.BAS" « auf Ihrer Eingabediskette. Dieses kurze Programm sorgt beim Booten des Computers dafür, daß das Prüfsummenprogramm automatisch geladen und ausgeführt wird. Achtung: Das Prüfsummenprogramm läßt sich nicht vom DOS-Menü aus starten! Die oben aufgeführte Programmzeile muß unbedingt beim Bootvorgang ausgeführt werden. Auf Ihrer Prüfsummer-Diskette sollten sich also folgende Files befinden:

- 1. DOS.SYS (entweder DOS 2.0 oder 2.5)
- 2. DUP SYS (passend zum DOS.SYS-File)
- 3. AUTORUN.SYS (Turbo-Basic XL)
- 4. AUTORUN.BAS (das Initialisierungsprogramm)
- PRUEFER.OBJ (das eigentliche Prüfsummenprogramm als Maschinencode-File)

#### Basic-Programme schnell und zuverlässig eingegeben

Um also, unter Turbo-Basic XL, den Prüfsummer verwenden zu können, brauchen Sie nur Ihre Prüfsummer-Diskette ins Laufwerk einschieben und ihren Computer einschalten. Alles weitere erledigt der Computer von selbst.

Vergessen Sie bitte nicht, sich von der neuen Prüfsummer-Version auch eine Sicherheitskopie anzulegen. Mit der alten Version sollten Sie unter keinen Umständen mehr arbeiten. Sind noch Fragen offen, wenden Sie sich bitte an die Redaktion. (wb)

```
100 REM *****************
                                          <LU>
110 REM #
                                          <HA>
120 REM #
            PRUEFCODE fuer ATARI
                                          (CII)
130 REM #
                                          (HE)
140 REM #
                W.Kress
           von
                                          (ZZ)
150 REM .
                Lindenweg 17
                                          <XA>
160 REM #
                7590 Achern
179 REM
       *
                by Happy-Computer
                                          <DC>
190 REM #
                                          <HD>
198 REM
        ****
                                          <HH>>
200 REM
                                          <AJ>
210 DIM F$(15),SED$(2),D$(504),C$(11)
                                          <RC>
220 C#="{CTL , ) (CTL D) U7 (CTL A) (CTL F) ) <
(CTL M) (CTL B)8"
                                          (IZ)
230 GRAPHICE 0: POKE 82,0: POKE 752,1
                                          (KV)
240 POSITION 12,10:POKE 201,8
                                          (BD)
258
   ? "Bitte_warten_!":POKE 752,0
                                          < DX >
260 Z=1000
                                          (62)
270 FOR I=1 TO 497 STEP 8
                                          (OL)
280 PSUMM=0
                                          <BM>
290 FOR J=0 TO 7
                                          (PK)
300 READ SED$
                                          <ZV>
310 H=ASC(SED$(1))-48
                                          (EQ)
320 IF H>9 THEN H=H-7
                                          KLT>
330 L=ASC(SED$(2))-48
                                          (GE)
340 IF L>9 THEN L=L-7
                                          (SR)
350 DEZ=H*16+L:D*(I+J)=CHR*(DEZ)
                                          <JV>
```

```
368 PSUMM=PSUMM+DEZ
                                                 <MV>
378 NEXT J
                                                 <0R>
380 READ ZSUMM
                                                 <AW>
390 IF PSUMM=ZSUMM THEN 410
                                                 < IN>
400 ? "(ESC CTL 2)DATENFEHLERA!":LIST ZI
END
                                                 <YX>
410 GSUMM=8SUMM+PSUMM: Z=Z+10
                                                 <N3>
428 NEXT I
                                                 (FY)
430 READ ESUITE
                                                 (SL)
440 IF ESUMM-BSUMM THEN 460
                                                 <Z0>
450 ? "(ESC CTL 2)FEHLENDE_ODER_DOPPELTE
_DATA-ZEILE":END
                                                 <IZ>
468 ? "(ESC CTL 2)"
                                                 <EX>
470 7 "(ESC CTL <)"
                                                 <RP>
480 POSITION 2,6:? :POKE 752,1
                                                 <UD>
490 7 ,"(CTL Y)1(CTL Y)_Auf_DISKETTE.sch
reiben(ESC CTL =)(ESC CTL =)"
                                                 <AY>
500 ? ,"(CTL Y)2(CTL Y)_Auf_CASSETTE_sch
reiben(ESC CTL =) (ESC CTL =) "
                                                 <A@>
518 ? . " (CTL Y) 3 (CTL Y) _ ENDE"
                                                 <VR>
520 OPEN #1,4,0,"K":GET #1,K:CLOSE #1
530 K=K-40:IF K<1 OR K>3 THEN ? "(EBC CT
                                                 <NQ>
L 23*1:60T0 520
                                                 <WU>
540 POSITION 3,4+3*K1? "--->";
                                                 <FU>
550 POSITION 2,20:? "(ESC SHIFT DEL)";
                                                 (EW>
560 ON K GOTO 578,598,680
578 POKE 752,8:? "{ESC TAB}DATEINAME.:
                                                 <DI>
```

Listing zum Atari-Prüfsummer



#### D1: (ESC CTL +) (ESC CTL +) (ESC CTL +) (ESC CKM5 CTL +>" 580 POKE 764,44: INPUT F#: D=1:GOTO 610 <ZI> 598 F#="C:":D=0 <MT> ,"CASSETTE\_FERTIG\_MACHEN, \_RETURN"; CHAN 610 TRAP 668 <0.1> OPEN #1,8,128,F\$ CYMO A28 630 IF D THEN ? #1;D\*(1,500); 640 IF NOT D THEN ? #1;C\*;D\*(7,494) **<UB>** <XK) 650 CLOSE #1:GOTO 470 (EW) BAA "(ESC CTL 2)(ESC CTL -)(ESC SHIFT DEL) (ESC TAB) I/O\_-\_FEHLER\_"; PEEK (195); CCRO 670 TRAP 40000: CLOSE #1:GOTO 480 (HQ) 480 POKE 752,0:END <YW> 690 REM <ZP> 1000 DAYA\_FF,FF,00,38,E7,39,A2,FF,1271 (EA) 1010 DATA\_BD,EB,38,70,00,06,CA,E0,1069 (NE) 1828 DATA\_FF,D0,F5,A2,60,BD,8A,38,1349 <VW> 1030 DATA\_90,00,01,CA,10,F7,A2,0F,800 < DL> 1040 DATA\_BD,00,E4,9D,61,01,CA,10,890 **31 15** 1050 DATA\_F7,18,AD,65,01,69,01,BD,793 (MP) 1060 DATA\_68,06,AD,66,01,69,00,8D,635 (CE) 1070 DATA\_6C,06,A9,ZA,8D,65,01,A9,737 <XW> DATA\_06,80,66,01,AD,67,01,80,668 1.080 (VG) (HH) 1090 DATA\_30,06,AD,68,91,8D,34,96,531 1100 DATA\_38,AD,30,02,E9,02,8D,02,657 (ZR) 1110 DATA\_06,8D,11,06,AD,31,02,E9,627 1120 DATA\_00,8D,07,06,8D,12,06,A9,488 (EV) < DM> 1130 DATA\_45,20,7A,38,90,11,8E,26,620 (CP) 1140 DATA\_06,A9,00,20,7A,38,90,07,536 < YO> 1150 DATA\_8E,17,06,20,01,06,18,60,330 (DZ) 1160 DATA\_A2,00,DD,1A,03,F0,08,E8,892 (CH) 1170 DATA\_E8,E8,E0,21,D0,F4,18,60,1293 (CK) 1180 DATA\_70,42,39,01,9E,06,9E,06,564 **(UJ>** 1190 DATA\_9E,86,75,86,9E,86,9E,86,615 (EX) 1200 DATA\_80,00,00,00,40,2E,22,54,240 CEYS 1210 DATA\_3A,22,20,50,45,45,48,28,469 <TN> 1220 DATA\_31,31,33,Z9,2B,32,35,36,390 1230 DATA\_2A,50,45,45,48,28,31,31,473 (LL) <SG> 1240 DATA\_34,29,9B,01,61,45,01,04,420 (SM) 1250 DATA\_54,30,72,75,65,66,63,6F,776 CVWS 1260 DATA\_64,65,00,1A,00,00,1E,00,257 (ER) 1270 DATA\_00,10,00,00,08,63,09,00,144 (LV) 1288 DATA\_18,15,00,62,79,00,37,0E,333 <NU> 1299 DATA\_00,28,72,65,73,73,0F,77,622 (KV) (OK) 1300 DATA\_62,68,A9,FF,BD,30,02,A9,986 1310 DATA\_FF,8D,31,02,A0,03,89,00,795 1320 DATA\_01,99,FF,FF,88,18,F7,A0,1223 (ZL) (DR) DATA\_FF, A2, 05, BD, 33, 01, 99, 1A, 842 <00> 1339 1340 DATA\_03,C8,E0,03,D0,02,A0,FF,1055 <00> 1350 DATA\_CA, 10, F0, 60, A5, 77, F8, 8A, 1088 (DT) 13AB DATA\_A9,FF,85,26,A9,FF,85,27,1191 (LE) 1370 DATA\_C6,77,A5,73,FØ,ZD,A5,A1,1208 CURS 1380 DATA\_18,04,06,73,F0,25,A6,78,896 (NG) 1390 DATA\_BD,14,01,E6,78,C9,98,D0,1124 < 2M> DATA\_4F,A2,00,86,78,86,73,A6,710 <WM> 1400 (QR) 1410 DATA\_A0,86,71,A6,A1,86,72,A2,1144 1420 DATA\_9E,86,26,A2,06,86,27,E6,901 1430 DATA\_77,D0,35,20,3E,F6,08,C9,929 (AY) CMD DATA\_98,D0,02,E6,73,28,60,A2,1008 <10> 1440 1450 DATA\_00,C9,98,F0,26,86,CC,A6,1138 **CLND** <025 1440 DATA\_00,86,D1,85,CD,A0,08,46,919 1470 DATA\_D1,90,0C,18,A2,03,B5,CD,940 (DL) 1480 DATA\_75,CA,95,CD,CA,D0,F7,06,1336 <ZZ> DATA\_CD, 26, CC, 88, DØ, E9, E6, 00, 1254 (EW> 1492 1500 DATA\_A0,01,60,A9,A4,A0,02,20,784 (HA) DATA\_CC,06,A2,03,B5,CA,95,CD,1112 (MP) 1510 1520 DATA\_CA,D0,F9,A9,1A,20,CC,06,1096 (IN) 1530 DATA\_A5,CD,18,69,21,8D,48,01,746 (BX) 1540 DATA\_A5,D0,69,21,8D,47,01,84,856 (UD) 1550 DATA\_D0,C8,84,00,60,85,72,84,1015 **KLY>** 1560 DATA\_71,86,CB,86,CC,86,CD,A0,1287 **<**JS> 1570 DATA-18,18,A2,05,36,CB,CA,10,690 (RJ) 1580 DATA\_FB,38,A2,03,B5,CA,F5,6F,1211 <RY> 1590 DATA\_95,73,CA,D0,F7,90,0B,B5,1257 **(GI)** 1600 DATA\_74,95,CB,E8,E6,03,D0,F7,1382 CIII 1610 DATA\_E6,D0,89,D0,DC,60,E2,02,1326 CVES 1628 DATA\_E3,82,00,38,80,00,00,00,285 **<8R>** 1630 DATA 53423 <NQ> Listing zum Atari-Prüfsummer (Schluß)

# Farbe auf die Fläche

Das normale Atari-Basic kennt leider keinen Befehl, eine Fläche einzufärben. Das XIO-Kommando ist nur ein schlechter Ersatz. Das Programm »Fill« bringt Farbe auf die Fläche und schraffiert sie sogar.

Das Maschinensprache-Programm »Fill« besteht aus zwei

 Dem Hauptprogramm (333 Byte lang), das in einen String geladen werden muß (beim Demoprogramm MA\$).

 Den Unterpgrogrammen (86 Byte lang), die in Page 6 von der Adresse 1536 bis 1621 stehen.

Bevor »Fill« gestartet werden kann, müssen die Zahlen aus den DATA-Zeilen geladen werden. Nach dem Einschalten von »Grafik 8« wird die Figur gezeichnet, die ausgefüllt werden soll. Um die Fläche festzulegen, wird mit »COLOR O« und »PLOT X,Y« ein Punkt auf die Stelle gesetzt, von der aus das Fill-Programm starten soll. COLOR O ist deshalb wichtig, damit der Punkt unsichtbar bleibt und beim Ausfüllen nicht als Hindernis erkannt wird

Danach wird das »Fill«-Programm mit »Q = USR(ADR(MA\$), MUSTER)« gestartet, »ADR(MA\$)« gibt die Anfangsadresse von MA\$ im RAM-Speicher und damit auch die Startadresse des Maschinensprache-Programms an. »MUSTER« kann eine Variable oder einen Wert zwischen 0 und 255 sein. Hiermit wird das Bit-Muster angegeben, mit dem ausgefüllt werden soll. Der Dezimalwert 255 ergibt als Binärzahl 11111111, die sich wie folgt errechnet:

 $1x2^{0}+1x2^{1}+1x2^{2}...+1x2^{7}=255.$ 

Hier wird jeder Punkt gezeichnet. Bei Dezimalwert 170 (entspricht binär 10101010) wird nur jeder zweite Punkt auf dem Bildschirm sichtbar. Das Bitmuster wird in jeder Bildschirmzeile um einen Punkt verschoben, so daß sich daraus schräg schraffierte Flächen (von links oben nach rechts unten) ergeben. Allerdings muß die Variable »MUSTER« dann ungleich 0 sein.

Um die Funktion des Programmes zu verstehen, ist es wichtig, etwas über den Bildschirmspeicher in Grafikstufe 8 zu wissen. Die Anfangsadresse des Bildschirmspeichers steht in den Adressen 88 und 89. Sie wird von Basic aus mit »PRINT PEEK(88)+PEEK(89)\*256« errechnet.

In Grafikstufe 8 ist der Bildschirmspeicher aus 192 Zeilen mit je 40 Byte aufgebaut. Jede Zeile ist auf dem Bildschirm einen Punkt hoch. Ein gesetztes Bit aus einem Byte bedeu-

PROG	RAMM-STECKBRIEF
Programmname	Fill-Routine
Programmtyp	Grafik-Utility
rogrammiersprache	Basic und Maschmensprache
rogrammlänge	3365 Byte
ür Computer	600XL/800XL/130XE
usätzliche Hardware	keine
Eingabehilfe	Prüfsummer
Bernerkung	zum Füllen von Flächen in den Grafikstufen 4, 6 und 8
eserservice	Diskette (FILLBAS)

tet, der Punkt ist sichtbar. Pro Byte können also acht Punkte gespeichert werden. Bei 40 Byte pro Zeile mit je acht Punkten kommt man wieder auf 40 x 8 = 320 Punkte Auflösung. Schreibt man in die erste Adresse den Wert 128, erscheint links oben ein Punkt. Addiert man zu der Adresse 40, wird der Punkt eine Zeile tiefer gezeichnet.

Das Programm »Fill« errechnet zuerst aus der Grafikcursorposition, die durch »PLOT« gesetzt wird (Adresse 90 für vertikale und 91/92 für horizontale Position), die Speicheradresse
im Bildschirmspeicher. Dieser Wert wird in XA, YA und XS, YS
gespeichert (in Maschinensprache gibt es keine Variablen;
hier müssen freie Speicherzellen verwendet werden). XA und
YA sind die Koordinaten des Arbeitspunktes, der durch das
Programm verschoben wird. XS und YS werden, nachdem die
erste Hälfte der Fläche ausgefüllt ist, dazu verwendet, den
Arbeitspunkt wieder auf die Ausgangsposition zu setzen.

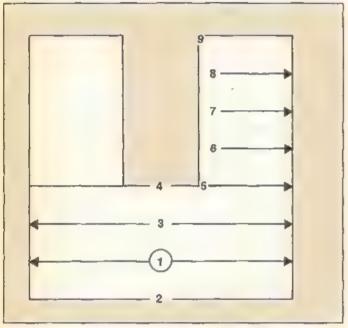


Bild 1. Verschiebung des Arbeitspunktes beim Ausmalen einer Fläche. Bei diesem Beispiel erfolgt das Füllen der Fläche rechtsbündig. Um auch den linken Teil füllen zu können, muß der Arbeitspunkt in die noch leere Fläche gesetzt werden und das Programm»Fill« nochmals gestartet werden.

Vom Arbeitspunkt aus zeichnet das Programm so lange eine Linie nach rechts, bis es auf ein Hindernis stößt. In Maschinensprache heißt das, daß von der Position XA, YA aus jedes Bit im Bildschirmspeicher abgefragt werden muß. Wird bei der Abfrage ein bereits gesetztes Bit gefunden, stellt dieses den Rand der zu füllenden Fläche dar. Der Füllvorgang wird dann in umgekehrter Richtung fortgesetzt. Das gleiche gilt, wenn der Bildschirmrand erreicht ist. Findet das Programm auch hier ein Hindernis, wird YA um 1 erhöht (der Arbeitspunkt wird um eine Zeile nach unten verschoben). Ist ein Hindernis direkt auf dem Arbeitspunkt XA, YA, kann weder eine Linie nach rechts noch nach links gezeichnet werden. Der Arbeitspunkt muß dann so weit nach rechts verschoben werden, bis eine freie Stelle auf dem Bildschirm gefunden ist. Auch hierfür werden wieder die einzelnen Bits im Bildschirmspeicher getestet. Liegt der neue Arbeitspunkt noch innerhalb der auszumalenden Fläche, setzt sich das Ausfüllen der Fläche fort.

Liegt der neue Arbeitspunkt aber schon außerhalb der auszumalenden Fläche, wird er so lange nach links verschoben, bis eine freie Stelle gefunden wird. Bild 1 verdeutlicht diesen

ı				
ı	XS	= 90	\$5A	
	XS	- 91	\$5B	Startposition
Ì	Brt	= 92	\$5C	
ı	YA	= 86	\$60	
ı	XA	= 97	\$61	Arbeitspunkt
l	BitA	= 98	\$62	
l	Rand R	= 100	\$64	rechter Rand der Fläche
١	Rand RB	= 101	\$65	TECHTOR TABLE GOT TRACTIC
ł	Rand L	= 102	\$66	linker Rand der Fläche
1	Rand LB	= 103	\$67	Illinoi Harid dei 1 lacito
ı	Lücke	= 104	\$68	
ļ	Sicher	<b>= 105</b>	\$68	kurzzeitiger Speicher für
ŀ				Zwischenergebnisse
ŀ	Modus	= 107	\$6B	
١	Blld	= 108,	\$6C,	Adressiert die Stelle im
١		109,	6D	Bildschirmspeicher, an der gerade gearbeitet wird
-	Muster	<b>= 110</b>	\$6E	wird über den USR-Befehl gesetzt
1				

Bild 2. Verwendete Adressen

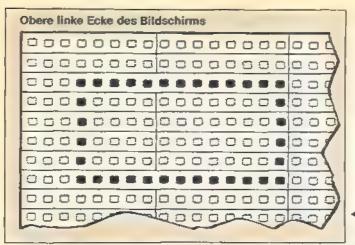
Vorgang. Ist auch sie schon außerhalb der auszufüllenden Fläche, ist deren unteres Ende erreicht. Der Arbeitspunkt wird jetzt wieder auf den Ausgangspunkt gesetzt (XA=XS; YA=YS). Dann wird die Fläche wieder ausgemalt, nur das XA nach jeder fertigen Zeile nicht um 1 erhöht, sondern erniedrigt (eine Bildschirmzeile nach oben). Kann auch hier der Arbeitspunkt nicht mehr innerhalb der auszumalenden Fläche verschoben werden, oder ist der obere Bildschirmrand erreicht, wird ins Basic zurückgekehrt.

Mit der »Fill«-Routine können fast alle Flächen ausgefüllt werden. Teilt sie sich aber in vertikater Richtung (zum Beisplel bei einer U-Form wie in Bild 1), wird nur einer der Arme ausgefüllt. Hier muß der Ausgangspunkt mit »COLOR 0« und »PLOT X,Y« in den noch nicht ausgemalten Bereich gesetzt werden und das Programm »Fill« muß noch einmal gestartet werden.

Eine Fläche mit mehreren Verzweigungen in horizontaler Richtung wird jedoch schon beim ersten Mal korrekt ausgefüllt. Das Programm arbeitet normalerweise nur in Grafikstufe 8. Es kann aber auch für Stufe 6 oder 4 umgeändert werden, da sie fast den selben Bildschirmspeicheraufbau besitzen. Dafür müssen folgende POKE-Befehle an einer Stelle nach dem Einlesen von »Fill« aus den DATA-Zeilen in das Basic-Programm gesetzt werden: A=ADR(MA\$)

Grafikstufe	4	6	8
POKE A+71,	48	96	192
POKE A+153	9	19	39
POKE A+260,	9	19	39
POKE 1552.	10	20	40

Positionen werden bei »Fill« in einer 1-Byte-Adresse für die vertikale (y-Achse) und in einer 2-Byte-Adresse für die horizontale Richtung gespeichert. So bekommt man mit »PRINT PEEK(84)« die momentane Zeile und mit »PRINT PEEK(85)+(PEEK(86)\*256« die momentane Spalte des Grafikcursors. Mrt »PRINT PEEK(90)« erhält man die vorhergehende Zeile und mit »PRINT PEEK(91)+PEEK (92)\*256« die vorhergehende Spalte, in der sich der Grafikcursor befunden hat. Diese Adressen, die als dezimaler Byte-Wert gePEEKt werden, werden auch von den Basic-Befehlen DRAWTO und XIO 18 (Füllen von Flächen) verwendet. In Bild 2 sind noch alle von »Fill« verwendeten Adressen aufgeführt. Bild 3 und Bild 4 verdeutlichen noch den Aufbau des Grafikbildschirms in Stufe 8. Die einzelnen Bildschirmdaten werden Zeile für Zeile im RAM-Speicher abgelegt. Um die Startadresse des Bildschirmspeichers zu bestimmen, müssen die



Adressen 88 und 89 abgefragt werden. Geben Sie hierzu bitte folgende Zeile ein: \*PRINT PEEK(88)+PEEK(89) \*256«.

Das Programm benützt zwei Koordinatenpaare zum Speichern von Positionen auf dem Bildschirm:

XA, YA = Arbeitspunkt: Dieser Punkt wird vom Programm geändert. Von hier aus werden im Programm Linien nach rechts und nach links gezeichnet.

XS, YS = Startposition: Dieser Punkt wird nicht geändert und zum Setzen von XA und YA verwendet. Die Startposition errechnet sich aus der Position des Grafikcursors.

(Rolf Kilian/wb)

 Bild 3. Ausschnitt des Bildschirms in Grafikstufe 8. Die ausgefüllten Punkte sind sichtbar, alle anderen Punkte auch Pixel genannt — sind unsichtbar.

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
00000000	00000000	00000000	0000000	00000000	00000000	0000000	วอบอ
Oberste Bildschi	irmzeile (Zeile 0)					un	d so weiter
Byte 40	Byte 41	Byte 42	Byte 43	Byte 44	Byte 45	Byte 46	Byte 47
00000000	00000000	00000000	00000000	0000000	0000000	0000000	0000
						un	d so weiter

Bild 4. Jede Bildschirmzeile in Grafikstute 8 setzt sich aus 40 Bytes mit jeweils 8 Bits zusammen. Daraus ergibt sich die horizontale Auflösung von 320 Punkten. Der Bildschirm setzt sich aus 192 solcher Zeilen zusammen.

10 REM Demoprogramm zu "FILL"	<rt></rt>	320 PLOT, 220, 100:0=USR (ADR (MA\$), 255):REM	
O REM *********	<3B>	*** Starten von Fill; Muster=255 -> Fla	
STO REM ROLF KILIAN	<hx></hx>	eche ganz ausfuellen	<ts< td=""></ts<>
10 REM Deschwende 11	<fq></fq>	330 PLOT 175,165: Q=USR(ADR(MA\$),255)	<pi< td=""></pi<>
60 REM 7900 Ulm-Lehr	<cl></cl>	340 PLOT 145,165: Q=USR(ADR(MA\$),255)	<01
20 REM Tel. 0731/60591	<wz></wz>	358 R=95	KA
O REM ********	<jg></jg>	360 COLOR 1: GOSUB 500	<b>&lt;</b> SI
80 REM Oktober 1984	<on></on>	378 COLOR 8:PLOT 50.100:Q=USR(ADR(MAs).1	
NO REM	<jw></jw>	76)	<f< td=""></f<>
00 DIM MA\$ (333):REM Das Hauptprogramm w		380 PLOT 300.100: Q=USR(ADR(MA4).170)	< Z
rd in MA\$ gespeichert	(HR)	398 6010 398	< 120
10 DEG	<wc></wc>	499 REM *** KREIS ZEICHNEN	< Q:
20 A=ADR(MA\$): IF PEEK(A)=104 AND PEEK(A	(MC)	500 FOR ALFA=0 TO 360 STEP S	<u:< td=""></u:<>
	(00)		
332)-166 THEN GOTO 200	<0C>	518 X=XP+COS(ALFA) *R: Y=YP+SIN(ALFA) *R	< X(
25 ? CHR\$(125); "BITTE_WARTEN"	<gt></gt>	520 IF ALFA=0 THEN PLOT X,Y	<ti< td=""></ti<>
30 REM *** Zeile 140 kann nach dem 1.fe		530 DRAWTO X,Y	< ₽
lerfreien Lauf geloescht werden	<xs></xs>	540 NEXT ALFA	< 14
40 RESTORE 30000:FOR I-0 TO 418:READ P:		550 RETURN	<h< td=""></h<>
C+P:NEXT I:IF C<>50795 THEN ? "FEHLER."		30000 DATA_104,133,105,104,104,133,110,7	
N_DATAZEILEN": END	<fc></fc>	2,162,3,102,92,102,91,102	<y< td=""></y<>
50 RESTORE 30000	<bd></bd>	30002 DATA_105,202,208,247,162,4,38,105,	
60 FOR I 8 TO 332: READ P: POKE ADR (MA#)+		202,208,251,165,105,133,92	<t< td=""></t<>
.P:NEXT I:REM *** Hauptprogramm ginlese		30004 DATA_133.98,133,103,133,101,165,91	
	<ua></ua>	.133,97,133,102,133,100,165	<z< td=""></z<>
70 RESTORE 30046	(HR)	30006 DATA_90,133,96,169,1,133,107,208,7	
180 FOR I=0 TO 85: READ P: POKE 1536+I.P:N	21.017	1,176,18,169,0,24,102	<r< td=""></r<>
EXT I:REM *** Unterprogramme einlesen	<ly></ly>	30008 DATA_110,106,5,110,133,110,230,96,	111
199 REM *** Pyramiede aus Kreisen zeichn	(1)		<pi< td=""></pi<>
•	<bj></bj>	165,96,201,192,208,234,104	\F
en a constitute a		30010 DATA_133,110,165,92,133,98,133,103	
208 A=0:GRAPHICS 8+16	<ll></ll>	,133,101,165,91,133,97,133	<e< td=""></e<>
?10 R=20:5=30	<rp></rp>	30012 DATA_102,133,100,165,90,133,96,169	
220 FOR YP=30 TO 150 STEP 30	<xe>&gt;</xe>	,0,133,107,165,96,201,0	< 31
30 FOR XP-175-YP/2 TO 145+YP/2 STEP 30	<jm></jm>	30014 DATA_240,17,198,96,169,0,6,110,42,	
240 COLOR 1: SOSUB 500: REM *** Kreis zeic		5,118,133,118,298,5	<p< td=""></p<>
nen	<6R>	30016 DATA_176,2,144,233,96,32,0,6,166,9	
250 COLOR 8:PLOT XP,YP:REM *** Ausgangsp		8,32,54,6,133,104	<i< td=""></i<>
sition fuer Fill	<gi></gi>	30018 DATA_164,97,49,108,162,0,42,176,28	
260 Q=USR(ADR(MA\$),RND(0) +256):REM *** F	(41)	.232,224,8,208,248,32	<b>&lt;</b> J
200 R-CSK (HDK (IMS) , MID (0) #2507 : NEI *** !	<rh></rh>	30020 DATAL39,6,192,39,240,30,200,177,10	
ill mit zufaelligem Huster starten	<tw></tw>		<y< td=""></y<>
270 NEXT XP		8,298,234,32,37,6,288	1 4
280 NEXT YP	<ui></ui>	30022 DATA 242,165,107,209,140,240,204,1	
290 R-80:YP=95:XP=160:S=15	<va></va>	96,97,208,4,228,98,240,65	<t< td=""></t<>
300 CDLOR 1:60SUB 500	<se></se>	30024 DATA_32,70,6,32,41,6,134,101,132,1	
310 COLOR 0:PLOT 100,100:Q=USR(ADR(MA\$),		00,166,98,32,70,6	<r< td=""></r<>
255)	<va></va>	30026 DATA_133,104,164,97,49,108,162,8,1	

Basic-Listing zu »Fill»



06,202,176,19,208,250,32	<vc>  </vc>	30042 DATA_234,196,102,208,6,228,103,240	
30028 DATA_39,6,192,0,240,16,134,177,128	1107		<bl></bl>
.208.236.32.39.6.208	<xk></xk>	30044 DATA,24,144,166	(SZ>
30030 DATA_242,32,54,6,32,41,6,134,103,1		30046 DATA.165,96,133,105,169,0,133,108,	
32,102,24,232,208,182	<mp></mp>	162,8,70,105,144,3,24	<cmd></cmd>
30032 DATA 56,176,250,144,141,224,7,240,		30049 DATA_105,40,106,102,108,202,208,24	
6,42,232,144,17,176,246	<cn></cn>	3,133,109,24,165,108,101,88	<bt></bt>
30034 DATA-196,100,176,26,192,39,240,22,		30050 DATA_133,108,165,109,101,89,133,10	
200,177,108,162,255,208,231	<bt></bt>	9,96,169,255,37,104,37,110	<vx></vx>
30036 DATA-196,100,208,4,228,101,176,7,1		30052 DATA_17,108,145,108,169,255,133,10	
32,97,134,98,24,144,214	<ng></ng>	4,96,169,255,224,0,248,9	<fc></fc>
30038 DATA_166,98,32,54,6,164,97,17,100,		30054 DATA,134,105,24,106,202,208,251,16	
162,8,196,202,144,16	<0Y>	6,105,76,169,0,224,0,240	<tm></tm>
30040 DATA_208,250,192,0,240,190,136,196		30056 DATA_249,134,105,56,106,202,208,25	
,102,144,185,177,108,56,176	<\rightarrow\rightarro	1,166,185,76	<uc></uc>

Basic-Listing zu »Fill« (Schluß)

### Mit »Happy-Mon« auf der Suche

Will man den RAM-Speicher durchforsten, sollte man einen Maschinensprache-Monitor verwenden, »Happy-Mon« ist sehr leistungsfähig und bietet alle Funktionen, die Sie brauchen.

ekanntlich dient der RAM-Speicher eines Computers dazu, Werte hineinzuschreiben und wieder zu lesen. Diese Eigenschaft nutzt auch der eingebaute Basic-Interpreter im Atari-Computer. Allerdings fällt es schwer, von Basic aus zu verfolgen, welche Werte wie und wo im Speicher abgelegt wurden. Mit dem Basic-Befehl PEEK kann man seine Neugierde zumindest teilweise befriedigen. Dabei erhält man einen Dezimalwert zwischen 0 und 255. Was fängt man aber damit an? Möchte man nämlich wissen, welchen Befehl der gePEEKte Wert ausführt, muß man eine Befehlscode-Tabelle zur Hand nehmen. Dieses Verfahren ist sehr aufwendig, und nicht selten schleichen sich noch Fehler ein. Hier ist >Happy-Mon< eine willkommene Hilfe, denn unseren Maschinensprache-Monitor lädt man einfach zusätzlich in den Speicher. Das Programm kann dann einen bestimmten Adressenbereich disassemblieren oder einen Hexadezimaloder Dezimal-Dump auf dem Bildschirm ausgeben. Tabelle 1 zeigt eine Gesamtübersicht der verfügbaren Befehle von

#### Hexdump- und Editierfunktionen

Der Befehl »Z Adresse« bewirkt die Ausgabe eines Hexdumps ab Adresse »Adresse«. Es sind beliebige Anfangsadressen zwischen dezimal 0 und 65535 zulassig. Bevorzugen Sie hexadezimale Zahlen, müssen Sie das Kennzeichen \$ verwenden. Dies gilt übrigens für alle numerischen Befehlsparameter.

Beispiel: »Z \$8000« gibt die ersten acht Byte des »HappyMon«-Objektcodes in hexadezimaler Darstellung aus. Anschließend erscheint die Frage »Continue (Y/N)« wobei Y für
»ja« steht. Betätigen Sie die Y- oder die RETURN-Taste, um die
nächsten acht Byte auszugeben. Mit N für nein oder der
BREAK-Taste übergibt man die Kontrolle wieder an den Eingabeeditor. Empfinden Sie die ständige Frage »Continue
(Y/N)« als überflüssig, fügen Sie nur ein N an den Befehl an.
Beispiel: Mit »Z \$8000,N« erfolgt eine ständige Ausgabe, die
nur mit CONTROL 1 oder mit der BREAK-Taste unterbrochen
werden kann.

Übrigens können Befehlszeilen beliebig viele Leerzeichen enthalten. Der Befehl »Z \$8000,N« hat dieselbe Funktion wie »Z \$8000 , N«. Die einzige Einschränkung: Zwei Parameter dürfen nur von einem Komma getrennt sein.

Ähnlich den oben beschriebenen Befehlen arbeitet auch der L-Befehl. Die Syntax ist identisch zum Z-Befehl. Zusätzlich wird nur noch die ASCII-Umsetzung der Byte ausgegeben.

Die bisherigen Funktionen dienen nur zum Lesen bestimmter Adressen. Mit »Happy-Mon« sind aber auch verschiedene Speicherbereiche veränderbar. Dazu wird bei der Ausgabe von Speicherinhalten vor jede Zeile das Größer-Zeichen (>) gesetzt. Dieses Zeichen stellt im Eingabemodus wiederum einen Befehl dar. Erhält nämlich der Editor eine Zeile mit vorgestelltem Größer-Zeichen, so werden die acht darauffolgenden Werte in den RAM-Speicher übernommen. Eine erfolgreiche Übernahme bestätigt »Happy-Mon« mit einem kurzen, tiefen Ton.

Wurde also ein bestimmter Speicherbereich mit Z oder L gedumpt, können Sie diesen mit den üblichen Editierfunktionen wie DELETE, BACKSPACE oder INSERT verändern. Anschließend kann die betreffende Zeile mit der RETURN-Taste in den Speicher übernommen werden. Versuchen Sie diese Funktion aber bitte nicht an Adressen, die »Happy-Mon« belegt.

Beim Editieren muß sich übrigens mindestens eine Hexadezimalzahl hinter dem Doppelpunkt befinden. Bedenken Sie aber, daß sich eine Hexadezimalzahl stets aus zwei Zeichen zusammensetzt (0 bis 9 und A bis F). Die Verwendung und Anzahl von Leerzeichen ist freigestellt.

#### Weitere Editierfunktionen

Eine besonders schnelle Eingabe von größeren Hexcode-Blöcken erlaubt die Funktion A. Der Befehl »A \$5000« bewirkt die Ausgabe von »> 5000:« auf dem Bildschirm, wobei der Cursor hinter dem Doppelpunkt positioniert ist. Jetzt können bis zu acht Hexadezimalwerte eingegeben werden. Mit RETURN erfolgt die Übergabe der kompletten Zeite in den RAM-Speicher, und die nächste Anfangsadresse wird ausgegeben. Mit BREAK verläßt man diesen Modus wieder.

Ähnlich arbeitet Funktion E. Zusätzlich werden hier die ASCII-Werte ausgegeben. Mit RETURN ist eine Zeile dann im Speicher.

#### Der Disassembler

Zum Disassemblieren stehen »Happy-Mon« zwei verschiedene Modi zur Verfügung. Den ersten Modus ruft der Befehl »D Adresse(,N)« auf. Hier erfolgt die Ausgabe in mnemonischer Form und in der Reihenfolge: Adresse, Hexcode und

Befehl	Erklärung
Monitorfunktionen	
<hexnum: bis="" h1="" h8<="" td=""><td>Editieren im Speicher</td></hexnum:>	Editieren im Speicher
L. Anfangsadresse (,N)	Hex- und ASCII-Dump
Z Anfangsadresse (.N)	Hexdump
A Anfangsadresse	Automatische Adressenausgabe
E Anfangsadresse	Automatische Adressen- und Dumpausgabe
Disassembler	
D Anfangsadresse (,N)	Disassemblieren
U Anfangsadresse (,N)	Erzeugen eines vom Assembler verstandenen Formats
Speicherzugriffe	
M Anfangsadresse, Neuanfang, Anzahi	Verschieben eines Speicherblocks
P Adresse, Byte	Single Byte POKE
V Adresse, Word	2 Byte-POKE (Low/High)
F Anfangsadresse, Anzahl, Byte	Füllen eines Speicherbereichs
T Anfangsadresse, "string"	Setzt einen String im ASCII- Format ein
Suchen im Speicher	
HB Anfangsadresse, Byte (.N)	Sucht em Byte
HW Anfangsadresse, Word (,N)	Sucht einen 2-Byte-Wert (Low/High)
Ein- und Ausgabe	
O #Filespezifikation	setzt beziehungsweise ändert den zweiten Ausgabekanal
N	Löscht den zweiten Ausgabekanal
R Sektornummer, Bufferanfang	Liest einen Sektor von Diskette
W Sektornummer, Bufferantang	Schreibt einen Sektor auf Diskette
Hexadezimal-Dezimal-Wandlung	
? Dezimalwert	Wandelt eine Dezimalzahl in eine Hexadezimalzahl um
? \$Hexadezimalzahl	Wandelt eine Hexadezimalzahl in eine Dezimalzahl um
»Happy-Mon« verlassen	
v	
X	Ruckkehr zum DOS
Q	Ruckkehr zum DOS Rückkehr zum Modul (falls vorhanden)
	Rückkehr zum Modul
Q	Rückkehr zum Modul
Q Verschiedenes	Rückkehr zum Modul (falls vorhanden)  Führt eine Maschinensprache-
Q Verschiedenes G Adresse	Rückkehr zum Modul (falls vorhanden)  Führt eine Maschinensprache- Unterroutine aus  Umschaltung zwischen normaler

#### Befehlsübersicht von »Happy-Mon«

Mnemonic. Die Zeilen beginnen mit dem >-Zeichen, so daß sich die Hexadezimalwerte wieder editieren lassen.

Der zweite Disassembler-Modus wird mit »U Adresse(,N)« aufgerufen. In diesem Modus werden zu den mnemonischen Codes keine Adressen und Hexcodes, sondern Zeilennummern ausgegeben. Den erzeugten Code kann man dann mit einem Assembler, wie beispielsweise MAC/65, übernehmen. Bytes, die sich nicht disassemblieren lassen, werden

Fehlermeldung	Erklarung
1	Falscher Hexadezimal-Wert
2	Falscher Dezimal-Wert
3	Byte/Zeropage erwartet
4	Angegebene Anfangsadresse oder erreichte Arbeitsadresse liegt im »gefährlichen« Bereich von \$D410 bis \$D7FF
5	Falls bei der Eingabe einer bestimmten Anzahl eine Null ein- gegeben wird, tritt diese Fehler- meldung auf
6	Unbekannter Befehl
7	Fehlerhafte Syntax in einer Hexadezimal-Zeile
8	Trtt nicht auf
9	Falscher Parametertyp
128-255	Betnebssystem-Fehler

#### Die Fehlermeldungen von »Happy-Mon«

PROGRAMM-STECKBRIEF					
Programmame	Happy-Monitor				
Programmtyp	Utility				
Programmiersprache	Maschinensprache				
Programmlänge	6407 Byte				
für Computer	600 XL/800 XL/130 XE				
zusätzliche Hardware Eingabehilfe	Diskettenlaufwerk				
	AMPEL				
Bemerkung	Die abgedruckte Version läuft mit DOS 2 0 oder DOS 2.5. Für DOS-XL- Besitzer ist eine zweite Version auf der Leserservice-Diskette vorhanden				
Leserservice	Diskette (MONITOR.COM/MONITOR1.COM)				

als ».BYTE« dargestellt. Beachten Sie im Zusammenhang mit der U-Funktion auch den O-Befehl.

#### Spezielle Speicherzugriffe Befehl

»M Anfangsadresse, Neuanfangsadresse, Anzahl« »P Adresse, Byte«

»V Adresse, Word«

»F Adresse, Anzahl, Byte« »T Adresse, "String"«

#### Erklärung

bewegt einen Speicherblock

Einzelbyte-POKE

Doppelbyte-POKE im Low/High-

Format

Füllen eines Speicherbereiches Speichert ab Adresse »Adresse« einen String im ASCII-Format ab

#### Suchfunktion

»Happy-Mon« emöglicht es, den RAM-Speicher nach Einzel- oder Doppelbyte zu durchsuchen Um ein einzelnes Byte zu finden, geben Sie folgenden Befehl ein: »HB Anfangsadresse, Byte (,N)«. Nun beginnt ab Adresse »Anfangsadresse« die Suche nach dem eingegebenen Byte, und gegebenenfalls wird die zugehörige Adresse im editierfähigen Format ausgegeben. Entsprechend verfährt der Befehl »HW Anfangsadresse, Word (,N)« beim Durchsuchen des RAM-Speichers nach einem Doppelbyte.

#### Ein- und Ausgabe

Alle auf dem Bildschirm ausgegebenen Texte lassen sich auch auf einem Drucker wiedergeben. Ein zweiter Ausgabekanal öffnet sich mit »O #Filespezifikation«, wobei Kanal Nummer 2 verwendet wird. Mrt dem Befehl »N« kann man den Kanal wieder schließen. Als zweiten Ausgabekanal sollte man niemals den Editor oder den Bildschirm anwählen.





#### Schaffen Sie sich ein interessantes Nachschlagewerk und gleichzeitig ein wertvolles Archiv!



Kennen Sie alle "Happy Computer". Ausgaben von 1985? Suchen Sie einen ganz bestummten Testbericht? Oder haben Sie einen Teil eines interessanten Kurses versäumt? Suchen Sie nach einer speziellen Anwendung?

Damit Sie jetzt fehlende Hefte mit «Ihrem« Artikel nachbestellen können, finden Sie auf diesen Seiten eine Zusammenstellung aller wesentlichen Artikel der Ansgaben 01 bis 06 und 08 bis

12/85.

Und so kommen Sie schnell an die noch lieferbaren Ansgaben: Prüfen Sie, welche Ausgabe in Ihrer Sammlung noch fehlt, oder welches Thema Sie interessiert. Tragen Sie die Nummer dieser Ausgabe und das Erscheinungsjahr (z.B. 2/85) auf dem Bestellabschruft der hier eingehefteten Bestell-Zahlkarte ein. Die ausgefüllte Zahlkarte einfach heraustrennen und Rechnungsbetrag beim nächsten Postamt einzahlen. Ihre Bestellung wird nach Zahlungseingang umgehend zur Auslieferung gebracht.

(the live law)	Thei Sens	/Hangaba
	Alcterior	
<b>Сотрин</b>	Aming — ein Theurenequipe werd Wirfelichinst.	9/10 14/11
	Agg: Lage goluntiot Der Jima/4- ist asciscit de	12/2
	Opendation other nesses lines and hern reverse PC	13/10
	Harmoning the Maring Control of Marini	11/6 M/10
e0	Asset; Large golumino; Doy (Plant) - Ass sacciated do ou und bout Proyects PC Haumonorposter (Manaforthrough des Malde, Zonssequentes (Canon (Dire documents QU) Absorbitos (Plant) Abso	6/3
	The Arm hand owner denser Hammer Offsteiner Management	39/3
	Manhorbetteb in day disa Neuron DFD-Programme für dem Spectrum Nullmodem sum Aulenacione	22 0
	House DFU-Programm für dem Spectram	23 .0 12/1
- ANDERSON	Pollinocycle Bayer Ambraction Ampy-Serbesburg pen Int Side 37 Software har num Historial That are very owns (Software Messhertes and Empland) Wittelman til, 170 Mark	14/13
	Software that turn blantarif	1000
	Tracing werton wats (Schoolder-Heubetten see England)	9) 2 5/9
racking.		27.2
oppr	Commodore Flurps and Trab generalist Descondenanteur's file den their MC-800	5/ 1 i
	Opus -Misses für den apertrant	-5:46
	Quica One Die Floppy Alternative (MCSIC)	30/4
William wing SX	Opes -Masses the denoperative (MSIC) Outce Ones: Des Flooper Alternative (MSIC) Mins-Lapatición fros til, 17,19 -44. Das Municarindes Variable CMS)	107/1
	Der Liber MEK von Philips kommt	50.1
	Pum-Gaparacon note on 1, 1, 29 sec. Das Massiraconides Visigna Cale of Des Auto-36EX von Philips humps (PVM) and MCXII- omputer as quadra Eug a completion Options viole Philips Flores Evisies Cashira, Cristiana, und Canas (Pores Evisies Cashira, Cristiana, und Canas)	141/2
	Flores Dieses Ganyo, Gnidstar and Canon)	23/5
		46/3
	Topreptd Fit METE	15/10
Schen	bit den Bogunden Tepputt auf Erfolgsbarn Bit har für den C 44	7179
	Bücher für den C 44 Bücher zur DFÖ	13178
	Bother in 1000 Bother tim Denken (III)	139/10
		100,14
	Managharickus This path use test stationer	13/9
	Die houselle Helmoomputer (Winter -ES)	1/2
	Penkaunteilung in Berun MXX was Tratepf	5. 24
	Hactree Ketma unit sparte Semmer-CES 686 Ted 2)	979
	Missaeburiches Composite Messe Xilia nach wie vor regiones Die horines in Nestroomporter Wittigle (35) Puntassatellung vin Bezus Mittig van 1922 193 Backer Krives und (poure Semmer-CS) 686 Ped 2) Blobbretones und (mythe Semmer-CS) 686 Europe der Krives und (mythe Semmer-CS) Sommer Core (35) 696 Wester Wittige CB Sommer Core 696 Wester Wittige CB Sommer Care 696 Wester CB Sommer CB Somm	B/4
	Sommer CED 965 Weight Welle in Chicago - Test 1	978
	Software Super Super of Control (St. 1975)	
a nejs	Künstische intelligens in Westbeden (A. Daroph)	2/12
Allejs	Management President Mids projectory	22/5
	Spierviews	
	Estativista David Cruse Ghostbusters Astor)	1775
	David widder Der Gradt-Großmanner Internew mit den «Print Derp» Machen	4/3
	Inc. Transel (Charrier Adam)  Lay Night (Vine-Prinadien) Microsoft)	44/6
	Kay Meta (Vine-Praudent Microsott)	30/9
	Barrist ratus (Caropelin R 6311 77)	20.70
Langent		200
uervarado.	Buffware-Term	27/4
deli-dimin.	Eas Teatrogrammer, dam inch tokas (Someovenil/C 00) Jane kontra Applementes Dies Detalles im Teat (STE 90, Gentes 10x, CS-8030) (Nachhall auf Sanie 148 ju 4/80) DWE 305; Someovenintii	4979
	Direc Despitate Les Teur (STX 90, Germins 19%, CR-8000)	1673
	The Residence	1075
	Can be de Verbindung (EP 22 EP +4 CKE 0)	3675
	Rempeter and Jesus, Mahristyneiter Call Confermation  Bertenboctenburben, Wie restructer (Ottowne Str.)	34/1  SU/4
	Genéral au Sante 140 in 4-98, DWR 2005, Septemberhall aum Heiselbergeries Lein Neueropries Lein heiße Verbundung DP 22, EP 44, DXL 07 Kompakt und die eines Mahranderster GAP Genetermand Bergesbogestanden, wirt geschrecker (Grammen 20) Schöfen Echnicht wirt abchaffen (Sabelau) 600(1) Zwen Dreitzer für den Achtender (FRLQ 46) CIP 900 CIPCQ Artan 500 ST Struker Hir und 12 Bim 54) Christian der Scholen in 12 Bim 54) Der Notat Nakasport Varzander (X-1) Zane-stelle von der nakasport (X-1) Zane-stelle von der	21/3
	Schöne Schuft and actualities Nacions Vergoeichstert Star	3779
	Scotterm mil purken Typen (Cabricle 9008)	136711
	2 was Drudker für den Schneider (NLQ 401 CP 100 CPC)	11278
omputes	Charges my hoterchers \$25 Charges \$40	30/6 33/2 32/9
	Computer der drinen Art 323 57 and C 136	3/1
	Des Musik Massino Tamaha (X4)	2674
	Car -next- Spectrum	31/1
	Ein -Einsteiger- aus Tarwin (STT-90)	16/2
	Enterprise of his North Way der Mephania (PHC H)	25/6 36. t
	Quantinosprung im Schneckentempe (Quidt Vermon)	180
	Epraenor dui Deutich Talens (Ce-Tec MEXI	1873
	Shares "Incoder (Share M. 1900)	20/10
	Spermum plus odes Spectrum matus	28/9
	SVI X'Press en starkes 3tuck gur un Gorff (MSX)	28/9
	En - Construções aux Tanetin (ET 46).  Laverstate ut va. Inche lebe des Nephanio (FHC 84)  (2007). Coltinadem Euratie y c. din Web des PCa  (2008). Coltinadem Euratie y c. din Web des PCa  (2008). Coltinadem Euratie y c. din Web des PCa  (2008). Coltinadem Euratie	CTRACE.
	YOUR Feminishes Badeculan JELE Computer)	20/2
	Was preschalach at many Hamanananadar	20/2 16/12 16/12
	Wet at wet. A has 500 ST + and 200 ST) Wile investigation on which lightness opposite 9 Malk Computer on Verilletel. 3-200-Erfahrengen (NCD-1-Huppy für Spectrum) Chieromet Stauttment	C20/3
ra fivector	3-2of-Erfahrengea (MCD-1-Fjoppy für Spectrum) (Discovery/Spectrum)	32/3
	Observer Properties  In unplainthe Plant Sportman — VIC 1841 Interfered  Lant Propey and Sportman — VIC 1841 Interfered  Lant Propey and Sportman Propeysystem (Vincenna Byenna)  Sportman Diskuttentraystem an Plant Jone  Dec Sportman Diskuttentraystem (Diskutser order Sprend)  Dec Sportman Diskuttentraystem (Diskutser order Sprend)	31/4
	Laut Froppy auf (SpoodDos plus 'C64)	462 83/8
	Specifican Delegation of Physics (Vincent System)	30/3
	VC 94 wird rur Beanfingpy	42/4
ocordar	Doc Spectrum Spenser (Daleareconter Spens)	38/1
P0	DET auch my dem 71 RS 21 av T 30 44.	36/8 25/5
	Dec Spectrum Epicinar (Daleareconder Special) Ent Indiger Specifier (Enter (Daleareconder Mic 3010) DFO and Find Con Till Sc 21, for 104 (As 10) Kommisjolishon and deem Spectrum	137 A 100/2
		1,01/2
philippes.	Corrowers some a Morie General balta	176/11
	Dat anders Weg Spectrum Tastaturi Fascination der Technok (Fascher Technol: Robertel)	44/11
		40, Z 37/9
	Flackbildscharts mr Schwieben (LCD) für Apple Bel	37/9
	Plackbildschurn mr Schwächen (LCD für Appla IIc.) Grafpeit Supergraßt für den Spectralis Kalest den Dieb "Alazmanlage für C 64, VC III.	20/2
	Flachbildscham mr. Schwiehen (LTD für Apple Bc) Craipell Supergradis für den Spectrum Raine den Deb (Alarmanlage für C 64, VC 38) Chien oder Taran? Voorn Command Modul/C 65)	16/3 29/2 40/10
	Place-balded from no Schwiechen (LDD für Applie Bic) Chalpeld Sepergmüß für dem Spectrum Raties dem Deits (Flatersmitten für C 64, VC 189) Ohren oder Faren Vorce Companie Models C 64) Pergutaren für MSEX (Forenz 1, 35-88) Filosophy (Sovicies im Vargaticksborn) Roberter Fürfenziege dem Zelomit (Parchastechmit)	20/2

Station Afth the Heimschmidgler (Feach Rober)
Tabellinedden für Genick-Courtee (Auch Malthafel)
Vers Paremary von Malth-Crebesser (Boschess Santafel)

Maharapi.	That the	/
	Sultivary-Yesta	
Paris and a	Ex. Textprograms, day such takes (Newsword / C.44)	7274
	Inna Kardini Anglideleti Ci	15/9
	Schreiben par Schneider (Vergleschman)	4 /6
	Salama ban obna Press	9615
	ins Tamer die 1-Depos-Terryerschestung (CPC 684) Terryerscheinung für schernaum (demos-sytem für 3650) base. Erweisenung mit Spraant (Autor Bess, 10° 64) base. von für spectrum	4579
lymcher.	Basic, Errorsecung zum Spartant (Autor Basic 17 64)	7074
	Bank, wir für spectrum	44/6 30()
	True Assertation for Auto Computer an Verglands. Foregraph to approxima S.P. 86-80 Established Re 200 STI	38 1
	Shoot Parral, over Marrodove-biddenshad (Spettress)	D6 47
	Lago for dee Ates 520 57	347.1
	Machine & Anne, an Sakhee Stack Maschinesa pricibe at Sepas Dauberal (CPC 464) Perronal Regio für den Annehot (TPC 465) Prinsvann Matter tongen C 64 menjang 68000	DIKE
	Party nat Pages für den Atan 520 57	22.48
	Schoolship was you managed to the sampling below	42 .0
	Policy West Control of the Control o	143/5
	West in Scrabper Specisin Schuler (S. C. 64)	16076
Philipps	Day Programm day Programmy modific (Programmy)	33/3
		42713
	Our Preve to operation Shift E. Day Westering the soluting and Manner (C 60)	130 1
	Self-ware-Enachery derwechenger/francis (Apple II)	27/2
Smith:	Apple-Gerille sare und fein (Danie Denne)	35,75
	Dan Many Johnst Farby and down Eddarbirm (Armic)	5072 5272
	Self E. Daya Weichburgh Weichelmung und Michigan CD Gelfmann-Kachbern Germendenpenfendert (Higher DI Applie-Gesführ zum und 5des (Deutie-Drume) Beweisstersterster (Erzel Diego – Derecktgerogenheitet Des Massa Jeaney Farbe und dem Midmichten (Applie) Greifer zu der dem Weissterster Statisch zu der Gestürfen Von der	20/3
	Kone Sudar ner Actiones Transcript Programmi	E 2.73
	Schooler Linerenant Continues Postal	4375
	Stille deem koymock programmen. Leangreen Poocall Scalimenters Eduar-rescalent (printermanies) Van Contin v. swazy av a (Linghton Begor and Supporgrafile on the C 64 gas recytoch version). Earness "San Tochtillen Dopingmet)	-
	60 Mar C 00 gas residence	31/2
bro		18673
	Coase: 64 De Solvese may Access Copplet Spectrum sel Dealt OFO Vergleschsteet	14975
Lateramonia e	Spectrum and Dealt (DFO Vergleschniest) Spectrum Sepretunden	3473
	Sharngraching	135710
diam'r.		17 65
donis Inhada	Man höre und stathe Geger & Shond Software/C 54) Schoolt dem Commodore Schooltgrogmanne en Verglotch	5475 9676
	Scholtman pur Toieten	180/20
	Spinio-Youige	
	Junatoh	40/5
	Arwann	545/9
	Archon D Adopt	100/3
	Anytom Arbioter Land	34671
	A View to a Kill	100710
	Selfriguer Wade of Mathyoth	195/9
	Roulder Deep.	125/2
	Bauery Next section bands	13075
	Caste of Tager Casteard	12472
	Court Tools	1467
	Cycloma S-Reg	52.76
	Down at Mychine	128/2
	Don't buy this	105712
	December & Revenche	61175
	Done-don Drag-menling	134/1
	Druconwoodd	14975
	Languages and La	19579
	Dec; Zone	165/1
	Chine	166730
	European	44. 4
	Falorenham (S)	-45 '5 -85 C
	Port: Itia One	40vW
	Fort tile One Final Brance Bounds	4611
	Proxime group to Hollywood Proxy Frank	96E/14
	Genusone Wattout	45.4
	Giorn'——	1667
	Chorchesters	138 3
	Chost Chaper Cloud Affection Cross Country Zond Ends	500075
	Gentralina	148 9
	Georg a Revenige	(50/6
	HI I O	14875
	Hyper Spore Hyper Spore i	14979
	Hyper Sports i	143/3
	Jump Jut Kanatolas	148/9
	Econody Astrophic	160712
	Xisgir leng Machell	143/3
	Mind Order Monetonii.	196/6
	Maste of the See	20,72
	Manie of the See.	165/6
	Matrix Day Munichadopte	5075 14176
	Mt Do	307/30
	Monator Trivia	168711
	Kick Faldo plays the Open Rightshade	(9.
	Rodes of Yeard	100/12
	Os Coors Tennin	350/5
	Print II QL-Chora	14675 E5176
	Racing Description Set	6.4
	Rawa	45/5
	Stacres on Fraculas Societ Sub	140/1
	Roc of cont's New (Solidate Charle II)	1634
	Rocadord's Ret (Soldet Dask II)	168441
	Rocafort's flot (Boidet Dash II) Roafor House Sharkoth House	16874 13870 12172
	Rocaford's Rect (Soldet Dash II) Rock's Roll	168441

Printered	Treat Solve	Assyste
	Spinite-Tunto	
	Skp Shot Software Star	182/6
	Spelubling Standard Stowers	142/3 145/4
	Starton Support Carner II Support Populine II	151/9 120/9 14./8
	The Ancient Act of Wag The Delias Quest The Dara listics The Pourh Protocol	149/5
	The Dain Russian The Fourth Pronocol	147/9 180/9 185/1
	The Hacabilium's Guide to the Calumy The Hacabilium's Guide to the Calumy	138/4
	The Hobbist The Little Computer People Projekt	170/12
	The Craft The Way of exploding Plac	168/10
	Tour de France Whote in the World is Carriera San Daego Whistier's Brother	170. L. 183. L1 141/3
	White replicating	164/12
	World Championship Bering Evolus	170/12 182/8
	Zyrdalalim	141/3
	Spicie Tops Absorbage in Wellinson	152/5
	Alpan 8 Aminon Angiqua	154/9 172 10 53/8
	& Figure at	47/4
	Arme Challenge Arme Tomb	147/4
	Axio: Tomb hand Mackpool	95/1
		13679
	Caverne of Ehafts Sunan de Africa Dallas Outer	155/6 155/9 154/9
		5476
	Dashi ng the Cambbeat Dashi ng the Cambbean Death in the Cambbean Deathfair i Revelue	156/9 172, 12 143/8
	Curain	464/9
	Everyode's a Wally Forbedden Forest	173/10
	Forest at World's Cad Forest at World's Cad Forest at World's End Prad	153./6 54/9 04/6
	Chostherium Chostherium	+40/3 142/4
	Ghortoness Grade in Street	152/8
	Hertsovand Nects of Term	172.12 156/9 273 2
	Heron of Lags.	173. 43
	Nobled Hulk	165/3 163/8 85 1
	Mazerts Bects for Son Walty fowers of Babylow	154/8
	Larateka	184/6
	KNight Loren Lode Runnet Lode Runnet Mgaz of the Sun	174/13
	Mass of the Sup Massurage Nashing from Andrikass	144/8
	Nijedskiegu Styre Andriktopde Mitodskiedow Mitods Holling	54/9 174/1 167/4
	Personal Advantumb	130/2
	Petial II	4774 4470 747 0
	Queen fite Tiess Salass West	154/8
	Seads of Egypt Seads of Egypt Seads of Egypt dee Gra Manion	173. / 1 174./2 82/5
	due Cita Manion Manion Stup of Doom	173 6
	Secretary of Chipmengus Challes	15379 1647B
	Setup Poker Settemer Calling	152/5
	Super Nuny	171 10
	The Duline Quett The Histoliuse's Diside to the College	15970
	The instructe The instructe The Ooses	73, 2 73, 1
	The Witness Tupe Macchine	(46.3 173.1
	Transi II Utuna II Utuna II	158/9 (26/2 451/5
	Thirms 31	18 45
	Urymor Veltalia Whater's Bookes	45679- 173. 1 14478
	Zerpt-kim Zers-Salahian Zers-	153/6 (42/6
		(73/10
Assendang	Alle Neuro (lahimman marang-Kagala/G e-0	67/5
	Alle Reune (fahrensunemrung-Engalar/G e4) (Alte Sevensuling G 64) Benesien Basat, gaan eintheti Sod- ware Basar 2 G 64 Daterbank mil Ensem "uppfl (C 65) Det Halywelle Ernes Jepiner (65)	67/3 86/1
	Dec Mally vaction Comes Innuism (MSIO) Dec operations and "elector Ltd 24-/2 perform)	10/10
	Des Hallywardie Komes Ironam (MSS) Des geschrift am "Bellen in d. Mr. Zepedram) Des Mars "narvenan betraps (Description) Narbball auf Josan in Dis 1988 (Description) Narbball auf Josan in Dis 1988 (Bauntile D. C. 64) Eurbold; nar Jannes-Bern (Description) Eurbold; nar Jannes-Bern (Description) Eurbold; nar Jannes-Bern (Description) Machalla auf Gere d'il gen 1988 (Mars) Machalla auf Gere d'il gen 1988 (Mars)	74/8
	De noturnelf Datenverwaltung (Mainfile II C 64) Einbuck ins Stummaken (Duamenmher/CPC 664)	83/3 86/5 89/3
	Machinal and Sede 85 at 8 125	16/3
	reactivate and before the great the Furthermore one size is angiographic (VC-200/Assem) Georgickie Fundation trip down Christianal date 64 Happyraviah, date Fratar wedoo Mutalianah (L.d.M./C.66) Nacchani, and Jones — Im R. 36	B0/1.
	Nacideal euf Seile - 1 m f. 755 Morne Decodes fill Fenhamateure (Specurary)	1774
	September 44 Center of the Content o	63/10 63/3 62/12
	Programme in Annie (pa) Cases (1 44) Suchen neur danies Disspressalting CPC (46) Suchhall an Suta (2 in 8 0)	83/4
	Rachhall au Jelie v 7 in 8 lb 7 in 6	87712
Gustilic	Appen to Italies-untile out down Drucker Beworte States out down C 64	105/12
	Beweege Graft mit der behehlen (GPC 664) Beweegung vom obten eige behehlen (GPC 664) Fachspeierwen A sie Grafikhoberung GPU Matenatischen (Gebertram) Grafik Windom bekommt Machtweiche (Gebertram) Grafik Windom bekommt Machtweiche (Gebertram) Grafik Windom bekommt Machtweiche (Gebertram) Grafike Windom bei der	24/10 66/3
	Fastry-niceren A http://www.fastry-niceren.com/ Grafit-rotterring file Mateundauchter (Spectrum)	89./3 /08./12 68. 10
	Western of Sons 80 m 78 Confuctor for recommendation (C. 54)	T3/6
	Cashing abor Apple (f) Rores Factory (C. 94)	58,2
	Harry Fanthary (199) Resource-Graftit for den Sportrum Schmidt Graftit and den Compiler (Ld.M./C 64) Schmidt Graftit and den Compiler (Ld.M./C 64) Schmidt Sports a. (190) Schmidt Sports and (190) Schmidt Sports and (190) Schmidt Sports and (190) Schmidt Sports and (190) Schmidt Sports	88/3 48/8
	September Spiniste at "Allen Apple-Complement September Schington (Craftic Palent C 50)	60/2 101/12
	Solar Parties Opertried Space Edular C 64 Spaces distinct grant material (C 64)	\$2/1 \$1/9
	Sprites pai Schware (CPC 464) Your Bild, zum Spote (C 64)	74/9 55/8
	Sprates pas Software (TPC 484) Your Bild, zone Spoto (C 64) Zawber dar Farbon no Phagpe Pointer (Ld.M./Abart) MacNull auf Serie Bf pr 5 76	83/3
		90/4 90/5
	Assistantian deep guntaman (a.d.a. Consist (p. 404) 24t Penteurum für Zense und Ellippe (CPC 464) Zykioose für Casiline and Mathematikus (C 46) Nechhall seit Sont 71 (n. 2 165) 36 Fashem in Canille o für Asati	60/10
Spant.		63/4
	Des Slave des Marines (C.64)  Desbur des Volkreffes (L.d.)4, (C.64)  Nachhall auf Seste , (T.e. 8/85)	61/3
	Neuthini and Sapin (Turni 1987) Neuthini and Sapin (Turni 1987) Dur rabendo Raudes (C. 64) Pitarrantoricholes Chi (16) Alega (68 EByte) Neuthini and Seste E. In 5 (68)	79/3 58/2
	Dan Aptarpares name territorial galacters (Igabota C 24)	60/1 79/6
	Gosponstergand att Schoolder (CPC 404) Earthbell and Secte 55 to 5/95	74/2
	Hote: Eart Erectury Abertenet in Segulad (\$4.86/C \$4) Eart Laire of Whith Extrem C \$50	80/6 52/13
	Earle Leiton Wester Screen C 640 Ehwick zur. hastriger Leitner (VC 30)	69/2
	Mit diam Appale and the Transachanic (Alatina Aphalasia)	100/4

rent.	That for	in/Hargaba
	Mit dam Ataci-Communic and Obserbs (Sand)	68/8
	Mile Woodshiet ins Mandwer sinden. Maleine unit Tücke (C. 64)	20/7
	Macalinire hus Labyranth (C 60) Macalini and Saith 137 in 6/05 Mountain (C 66)	BD/6
	Hachting (Spectrum) Hachting and Josep 85 m 5/85 Nimenmajohand (C-64)	75/6 22/1 73/3
		78/5 54/8
	Percho die Macht des Geures (C 84) Nachtall auf Sebe 80 zs 13-15 Ramsiabres mu dem Jorstok (Crives/C 64) Retter den Johnen Bann (Insekt dahman/C 64)	71/4 73/2
	Magalian and Same 70 to 12 (86)	700171
	Scharzbelut Anni Brilliu Vocuciu Hochwaner Arannon/Ld.M./C 84) Chur don Woltens (Propplanein/C 84) Wartsochupial (Spectrum)	75/ 64/10 73/11
	Oper don Wollow (Propplements/C 60) Wastenshipini (Spectrum)	104/3
tricks.	AMPEI getinen Licht für Atari-Maschinge-Programme Auf dem Laufenden mit beider Echtzehnlie (C. 64)	70/9
	Auf dem Leufenden mit einer Echtmarahr (C 64) Auf Trap gebeucht (CPC 464) Aufortar für Avan	73/12 49/9 66/2
	Ramic Integration (G 64) Black-Computation (Spinistrate) Not Make and Senter 80 ins 12/85	BIL/10
	Basic Plus Application Basic Envestment (Applie II) Basic C 64 peopet dis	96/2 79/11
	Bildscherminek für den Commodern 64	79/1
	Dyte-Chater (Spectrum)  Chase Memps medich institling (CPC 44-0)	91/6 61/6 67/9
	Dum-Generator file Apple III Debter Rim und herperiasen (Arnel-Elifi)	86/4 84/10
	Database Manusa Avergana (Amat-2888) Database Manusa Avergana (Amat-2888) Data news Charlemanesses air da (C-64) Data news Charlemanesses air da (C-64) Data news Charlemanesses (C-64)	07/13
		60/13 69/4
	Diskuttum aparam (G-64). Fachhall auf Sacha 40 in 11780. Dash-bally für die achaelle Millio (Amed). Dash- und DelG-Unjay für allie Amel-Computer.	60/9
	Disk histo für die ocheelie Hille (Amri) Disk wird DOS-Uniny für alle Amri-Computer	71/8 27/10
	Druc Tracim, the 1400( Ean sangue Gressons für dem C 64 (Longuezeum 64)	02/6 71/11 06/1
	Posterficial nor HELP & TRACE (VC 20)  Feneter Schristler (C 64)	Tarks
	Dies Tochn Na May.  Lin surges, Semich für den C 64 (Lopquizien 64)  Palaminist mir HELP & TRACK (VC 28)  Fentockfristen (C 64)  Fentockfristen (C 64)  Fentockfristen (C 64)  Find Labe (Epectrum)  Find Labe (Epectrum)  Find Labe (Epectrum)	99.75 69.73 86.73
	Fahko-nates out bet langer Betruffolgen (GPC 464) Fahkoal Managet für Gerumsdote 64	23/9
	County Harderpy in viorischer urbho (C 66) Kontoniet Spiecheroriettsching in 66)	61/3 61/3 67/6
	Florito Pignisa, Main III steed Plancks (Spectrum) Flakibi-instrum and bus singer in Refruitering on CPG 664) Flakibil Marsagor (Nr. Communider 64 Kantali Marsagor (Nr. Communider 64 Kantali Marsagor (Nr. Communider 64 Kantali Kan	94.7 s
	Manchesercode Roumen in Basis nespenses (CPC 494)	101/3
	Nachabas ad Seco 20 or 12/90	100/11
	Munic and Farbe (C 64)	8678 86712 8176
	Petro I Triction for security of the Checkstammer (C. 61). No meets fundamentarised deem Checkstammer (C. 64). No meets fundamentarised Appeals and Token this labe Abort Computer (C. 64). Prodes and Token this labe Abort Computer (C. 64). Programmers with the Checkstam (C. 64). Programmers with the Checkstam (C. 64). Programmers where less the generactic (CPC - 86).	84/2
	Passa Werkman the des Programmers (C 60)	BQ/6
	Programmenster leicht gemacht (CPC 484) Programmenschund übr den Spectrum	79/10 84/5
	Proteins set dem Dructer (CPC-46-5) BUGH-Desk ner Auer 60000.	26/8 110/11
	Programmaschern für dem Spectrum (EPC ump Programmaschern für dem Spectrum Prodriktig und dem Dructurg (EPC 46C) SCAD-Dmit (FAARS 600XL) Ran sen dem Unerpos (C 46C) Ransunding 6 FC 6 FO Nachhall net Serve ( 7 in 6/166	70/8
	Reachman act Server (* (n. 37-18) Rock to Rehadenia (* 68) 3000 berkele olsten eige (Schneiden) Schulderman (Palanaerten) (* 64) Schuld met der Einschuginer (* 64) Hachhall set Sogne der (n. 17-16)	64/1
	Schildermaid (Piakuschutz) C 91) Schluß mit der Eintebielbeit (C 94)	64/1 23/2 86/10 70/1
	Machhall sef Sone 80 m. 12766 Seeden wascho fieldachgraverschiebeng (CPC 460)	
	Machinal and Soons 80 on 13.746 Senden weards hildsuchursverschiebening (CPC 4640) Senden weards hildsuchursverschiebening (CPC 4640) Senden weardsten mid Problitismen beloogt (Bipotening) Sprachinan für Commendens Binnie (CP 46) Senden March für Geregen (CPC 464) Tanvers-Für Grund (CPC 464) Tenden such in Certification (CPC 464)	26/6 86/10 86/1 106/18 80/1
	Spectrum rapit & Therm Spectrum rapit & Commodorn-Basic (C 64)	8071 547E
	Machinal aux Sono 100 in 0/ml	
	Statement out the Arms	95/10 95/79 73/10
	Turwood 654 ms DIN Taxanus (CPC 866). Texto such to Gradionodus (Utari)	73/10 80/8
	Title & Tooles cond was don Schoolship	84/3 77/6
	Tupe and Garn Asac: Vanablerelacity to Asac (Anan)	94/6 80/2
	Variables/Transfer (Spectrum) Vorfarter Latechetz (C 84) Nochhall auf Seste 30 m 12/86	80/2 80/31 66/30
	Von Marchinencode men Beste, Percentidate III IIII	96/10
	Zeilenaltzobetik suf dem Schneider Zwas "GPEENS um schneilen Wedtstel (Spectrum)	73/4
	Who the Buildar in their herritor (Alam) Sedentalroberth and does Schlooder Seen of PETNS in administration by Actived (Spectices) 222 Leality Schmidtelme for Americky (C-64) 3 to told Manachimentonic Bouganin (Spectrum)	91/11
M*		26/2
	Ornothagen Dates am kabenden Bund Dates auf der echtsellen Schollee Flaggy gegen Kanene	80/9 38/8
	Salbet peachmaint ist halb peaped.  So asbesist das ,050-Laufwerk von Rind.	30/E 06/8
	Six fatest wited in Naturally) elia 341	34/8
	Spec becreation Endleshand Tipe Torin and Tedelinden Wahn in Library and Bis and Bysee	41/5 24/8
19 c	Faithmoneage — busses Feature for Computer	127/5 132/6 147/10
	Monager Bishing popiess, quintell and genomes. Surve along Sriver war Leaveler's Gasordractors' Districtions (Forker (Thermodractor)) Testwintelle St. Lathington	135. JO
	Terreporter für Fachdrucken Mit leinen Tönen (Torenstenbidrecinst) Schalle Hedel, spine Typen (Martin und Typensid)	145/10
inji		14/10
	Bedehing-warmering for SCS (CPC 404) CP/M — Lin Betreaberguless Fensies in the Zukneht Beste auf dem 620 ST Lego-Specieres oder wreethafte Albertaltyn	132/13
	TEX — Marchineneperche mil Somori	110/1 56/11 151/3
	Councils and the U Councilson Gloichealast	146715
	Segrate has control to parameter. Control on the Bookharen — Bit St. Bit	163711 167711
	terrent (Craffe-Tablerts) Was say uth a maintent Companier (Testinaren) Das Interface ROM und seine Natrung	4376
90		19674 6574
	Der den und seus RAM	135/2
	MSX Der Standatte unter der Lupe	146711
	ter getons Absentive Registra Automatica Message Schwististe Dengair. Angeltr MASX Dur Sundatur unter der Lupe Schwististellen was spand das engretische Sie historie der Spieder Bestehtlagen. Ströme und Kandde um Spiedtragen.	3874 3275 13876
	Strome and Kansas an Spectrum Von Traum ram (wincomputer (88000 Prosessor) Weethe Handcopy (Schmeider)	30/1. 24/12
	Project in manager processor of the property o	158/12
	3D-Co-Br	2714
	Alignmeiro Domon Dur Computer — En mediamer Tricinari	0.079
	Keine Asom vor DPU	163/3
	IC: Abuntumur Dubleng Adura Stania Premioni	161712
	Bits and Abwegges Computer als Boofts/gue Das Daumenlano für Heimcomputer	147/1.
	Zigor L' 64 i.m L' 225	130/9 51/11
	En teurns Verynigen (OFO-Kosten) Enterry Mane Wedzmunepus mit Compelengstalit	27/6
	Happy-Sportspanifilities Rouncomputer aux ownion Hand Mah. aix ou. Computer (Dec Commodore Story)	142/12 142/12 49/4
	Mahi ais ein Computer (Dio Commodore Story) MSX neoor Standard neoe Chencen MSX Software Es goèt voran	116/9 126/9
	MSX-Software Engobi votate Ranpkopuerur gospet dan Rust der Welt Software Stad) geschiepkt	12679 128710 151710
	Softwarm (Sual) geosphysic Softwarm-Peartness Softwarm-Volkerflor	23/8 144/12
	Software rom Sparrand	153/10
	Spiele auf der schwarzen Liste Vom Heimgempgter-Freuk zum EDV-Spossibiten Vom Hohby zum Geldrogen	160.11 35/2 33/2
	Von Abertson: om Abentsone za schreiben Wenn mai was schiefgeht	140/12
	Westerston, Fragen and Anteriries in a 128er	525 11

Stickwort	That	Leito/Augabo
	Zu viel Kontrolle 1985 – Das Johr der Messchalts	350/3/ 254/4
	N-	BUTN T
Logo	Tell . Der Einsting für Übertreiger Tell 2: Die Schlieferbes anzur landen	45: /4
		.53L/5
Pascal	Tell 4: Die Schnickerber wird erweichen Placea für Schnickerber wird erweichen Placea für Schnickerber bei 19 Placea für Moge Eopfa-Freil 1 Placea für Moge Eopfa-Freil 2 Placea für Moge Eopfa-Freil 3 Schneille Graftir für Alzei Compouer	85/8
	Pascal für kluge Köpfe/Teil 3	81/9 -81/10 124/11
Anna	Pascat für kinge Köpfe/Teil 3 Schnelle Grafik für Atazi Computer	24/ D
C 64	Rusik mit Poke und Peek/Teil I Mank mit Poke und Peek/Teil 2 Mank mit Poke und Peek/Teil 2 Mank mit Poke und Peek/Teil 4	54/3 53/4
	Musik and Poke and Pock/Teil 3 Mank and Poke and Pock/Teil 4	56/S 52/6
	Lemma See them Commoders 64 terms / Teil   Lemma Sie them Commoders 54 leterany/Teil 2	59/6 46/0
	Lemen See Davin Commoders 69 ireanen/Test 1	45.78
	Lerron Set Envis Commoders 64 retacent/Test 6 Lerron Set Bres Commoders 64 tennen/Test 5 Lerron Set Bres Commoders 64 tennen/Test 6 Lerron Set Bres Commoders 64 tennen/Test 7 Ques Fresk bres Krest/Test 1	59/9 45/10 56/1:
	Chair Flori kein Knier/Tell I	40/12
CPC 494	Kein Buch mit mehan Stepeta/Teil 1 Kein Buch mit mehan Sporota/Teil 2	128/6
	James and the property of the Communication of the	0576 6679
Hardware	Zagūbarwachung per Computat Tail I	188/4
	Zugüberwichung per Compiter/Teil 3	13.76
	Tamela	
	Anci 100 ST and Alemogram  Blicher was down Walteld (Schoolder)	80/18 88/18
	Dam User Pert publicus sacist auf (C 64)	547 1 4378
	Fabler in der Zoodrum Hardware Ordnust nicht das leiste Wort (CPC 464) Guse Verhendung mit dem Schnader (PC-Intestage)	36/9
	Lightshow ma dem Commodere 54	\$87.10 4475
	Multiplier für den förstickenschlub (Spectrum) Nachhall auf Soise 65 in 6/65	30/3
	Outer Verbendung mis dem Schmidder (PC-Dannilland) Leidslander und dem Germonden 81 Musikalant für dem (nyritidinasehlut (Spootrum) Nechhold eit Göber 87 in 7768 Mechholl seif Geber 77 in 7768 Mechholl seif Geber 78 in 7768 Mechholl seif Geber 78 Mechholl seif Ge	89710
	Nie weeder Anget Alamanlage C 84)	48/3 141/8
	Machinal auf Sone 80 m. 2785 Solveisen und waiten mit dem Atan (Schalitaierinde)	
	Schuler und walter mit dem Aten (Scheltiniechse) Schreiben mit Schreibmeschie-engesträt (C 64)	14770 B6/2
	Schreiber and Weiteln mit verde Strant Johnstättlichemen filberheiben ent Schleiben der Strant Johnstättliche Gestellte Schreiberheiben schlieben der Hilb Tibyoph Schreiberheiberschlaße, Kast J. (1867 Tibyoph Schreiberheiberschlaße, Allei J. (1867 Tibyoph Schreiber auf einen Purt II Septemb Aussiger übertamm) Spetem auf einem Purt Schreiber auf einem Schlieben einer Schreiber auf einem Schlieben Schreiber auf einem Schlieben schreiber auf der Schreiber auf einem Schlieben schreiber auf der Schreiber auch der Schreiber auf der Schreiber auf der Schreiber auch der Schreiber auch der	26/2 24/3 197 11
	Statemant street Part (7 Septembra Aureige/Spectrum)	24/2
	Verbesserie Cursoreleaning from Spectrum Zumi Jovetche für ein Hallshijs (CPC 484)	09/2
		31/6
TI 80/4A	Mayersbarsichess Erweitungen zum Ti 96/4A	60,1
James	Markithomu bi Atari	287 1
Attact ST Southway	Rund up day Atas jedo Mongo Software Androll compati Familiana By 75th and Socomia	657
Зриготем С 64	pero wonger surveive Paraphete Par 2001 and Spectrum Batatanus Par don Commodate 64 Det Cumpetor mit dem großen Zubehör Bund um dan Schnedor	49/ 86/4
CPC 464	Rused not day Schmoder	4878
Alcustichoppier Dracitor	Altunairoppies, preiswart wie noch tie Druckerpande	129.10
Monitore	Dructorparkis Nachmall str Seise 60 in 18/85 Machmall str Seise 60 in 18/85 Machmall str Seise 60 in 18/85	±36/8
Monte	Hachhall sufficies 60 in 12/65 Musicustware	dire t
Grafit Software	Bunks Dunks Engage Street (Configuration or ma)	45 20
	So was software (Harmanitware IQI Herry computer)	327 480712 3879
Specie Computer	Soliaden (Die neurise Progressum sind ihre Preme) Solvia: .e(tware (Teumer)ware (If Heimscomputer) Spiele sie dem Beuleaten (Cohertrotto) Solo) Walcher Computer Stin, Walthachtsleet?	3873 38712
	Watheringe	
Animi Anii Anii Anii Anii Anii Anii Anii Ani	Aistica Aphianit	100/1
Authoring Author	Bultarenters (Markhank)	42/3
ANSOREM	Bithoven-Fautoni Bithoven-Fautoni Duc Computer ain Renousemen	12076
Autral Autral		1591.6
Auku! Kulloreng	Dec schöuste Tital von 1884 Dec schöuste Tital von 1994	10071 38676
Authority Author	Designation of the Research Computer Legentre Deservois Computer Legentre Deservois Computer Legentre Deservois Deservois Computer Legentre Deservois Deservois Computer Legentre Deservois Computer L	17671 D 3078
Autorang	Happy Computer Lessewertherers	26/12
Author	the Einster (Die beere Anwendung) the Einster	14479 18779
Autral Autral	Egnit is det Knape Leescombige – Taschenrechner Probleme auf der Würselm	
Autruf Subbeater	Problems auf der Würselm Spiel des Jahres	178/11 846/4
Autrel	Spiel des latres Stein and des Gempeter Was seuests, was regele? Was gewinst des goldenen Besonstel	
Andorrag	Was gowinst den goldenen Besenstel	46/11 .79/11
	Landson State	B4 4:
	Atun Tips Autostat für VC 20 Realcode 2 für MZ-700	03/
	Residence 2 ftp: NOZ-700 Based-Speciation white Bodes (C 64) Based and Hitter-Craftic (C 64)	77/3 189/11 180/12
	Convendere-Ecial	1907 2
	Eingeboreste beum Spectrum speuchess.	a16/3
	Gedlightsurricite balas 23( 5)	84/6 85/4 103/1
	Fur vacer ma Gedichtusfücke baim 2K 51 geen 44 as RK 50 angepah 10thbulgenbliehn beim VC 50 LPRINT 111 Fahletteses Druden auch ohne EPROM	103/1 77/3 .09/12
	LPR(NT II) Fahlettosos Drugken auch ohne EPRON Protigme auf den stagen Zeden (C 64)	.09/1a .06 )
	Problems nu 800 KL	59- 2
	Sperce-Epitisten (C 64) States aus dem Commodere 61	110/3
	The fin Cont 1 Unvolkethedge Advece beam EX 51 VC 30 and Videolamers are Monitor	77.73
	I fell day A I f. a I	.0371

#### Auch die bisher erschienenen Sonderhefte können Sie jetzt direkt bestellen:

DESCRIPTION OF STATE OF CASE

Anwei für all	#HEFT 01/85: SPECTRUM ndungsbesogene Listings und Tips&Tricks e Spectrum-Fans.	
Eine F	Tille wertvoller Beiträge und Listings e Schneider Anwender	
		-
		_
_		_
_		_
		_
_		_
_		_
		_
		_
		_
		_
_		_

Am besten gloich mitbestellen: Die Happy Computer-Sammelbox

Für alle Leser, die »Happy Computer« regelmäßig kaufen, sammeln oder im Abonnement beziehen, gibt es ein interessantes Service-Ängebot: die Happy Computer-Sammelbox!

Mit dieser Sammelbox bringen Sie nicht nur Ordnung in Ihre wertvollen Hefte, sondern schaffen sich gleichzeitig ein interessantes und attraktives Nachschlagewerk.

Übrigens: Die Sammelbox ist nicht nur ein praktisches Aufbewahrungsmittel: Sie eignet sich auch hervorragend als Geschenk für Freunde und Bekannte

zu vielen Anlässen.



0000:FF	FF	98	80	99	99	4C	BF<71>
00008:83	48	48	C9	ØA	90	06	18<0B>
0010:69	37	4C	12	80	18	69	30<9A>
0018:4C	17	82	AZ	26	86	85	C9(2B)
0020:30	98	9C	€9	SA	90	ØD	C9(1D)
0078:41	90	24	C9	47	90	09	AØ(28>
0070:01	84	85	60	38	E9	318	6Ø<4D>
0038:38	E9	37	68	AB	88	84	86<07>
0040:BI	81	28	15	80	A6	85	EB(CB)
0048:00	90	86	CS	E6	86	4C	3A(D5)
0050:00	A4	86	BI	81	C9	9B	FØ<43>
0058:13	C9	20	FØ	ØF	C9	3A	FØ<17>
0060:0B	C9	20	FØ	07	68	68	A0 <e7></e7>
0048:01	40	SD	82	60	AZ	22	A1 <d2></d2>
0070:81	20	15	80	85	83	A5	85(57)
2078: DØ	ED	20	30	62	60	20	67 <d2></d2>
							85<4C>
0080:80	A5	83	ØA.	96	AG	ØA.	
0008:8B	20	67	820	A5	83	18	65(F3)
0070:BB	62	82	68	20	67	88	AS(4E)
0078:83	85	84	20	78	92	60	ZEKCAS
00A0:78	80	AS	83	85	84	20	78(9C>
08:8ADD	60	A9	00	85	83	85	84 <b3></b3>
00E0: A2	99	AI	81	C9	ZA CO	DE	ØC(68)
000B: A5	97	85	83	A5	98	85	84 <d2></d2>
0000:20	30	83	60	A1	81	C9	2B(EC)
00C8: D0	ØB	AS 4C	9E	85	83	35 36	9F<Ø2> BØ <cb></cb>
0000:85	84	-	BA	80	20		67<3F>
	86	02	Ø1	90	4C	4C 78	80 <c8></c8>
0000:80 0000:0		02	62	03 4C		80	C9(9A)
00F0:04	DØ	502 D00	40	99	8E	99	98<08>
ØØF 6: 04							80KC4>
0100:A5	84	AC DØ	SD Ø1	82 68	20 68	68 68	A0<09>
0108:03	4C	5D	82	A2	20	A1	B1(19)
Ø11Ø:C9	ZA	DØ	00	20	36	83	A5<87>
Ø118:97	85	83	A5	98	85	84	60<01>
0120:A1	81	C9	2B	DØ	SIC.	A5	9E<74>
0120:85	83	45	9F	85	84	20	38(C2)
0150:83	60	A5	81	05	F3	A5	B2<3D>
@138:85	F4	49	00	65	F2	18	20<48>
0140:00	DS	80	20	20	D2	09	80 <aa></aa>
0148:25	A5	D4	85	83	95	05	85 <ba></ba>
0150:84	AØ	99	81	81	C9	2E	90<5A>
Ø158: ØA	C9	39	92	02	DØ	04	CB <d1></d1>
0160:4C	4D	81	CB	98	18	65	80<4F>
0168:85	80	20	22	83	66	68	68<85>
Ø170:AØ	02	40	50	82	20	26	81<33>
2178:AS	84	DD	01	60	68	68	AB<81>
0180:03	4C	50	82	A2	(36)	A1	81 <a2></a2>
@188:C9	24	FØ	03	40	26	81	20<3A>
@190: 50	83	4C	A4	80	92	20	A1<42>
0198:81	C9	24	FØ	03	4C	6F	81<8A>
01A0:20	30	83	4C	F7	82	A5	83<45>
@1A8:1B	46	40	48	40	20	<b>Ø</b> 3	80<47>
0180:A5	83	29	@F	20	63	BØ	6 <b>0</b> <68>
0188:AS	83	65	8C	A5	84	85	83<10>
0100:20	AØ	B1	A5	80	85	83	20 <ec></ec>
Ø1C8: AØ	81	AD	9A	86	C9	E1	FØ<43>
0100:05	A9	FF	SD	23	02	60	A5 <ed></ed>
Ø108:83	85	04	A5	84	85	D5	20<42>
01E0: AA	D9	20	E6	DB	AR	80	B1<46>
Ø1E8:F3	85	99	98	48	A5	99	29<65>
Ø1FØ: 7F	20	17	82	88	AB	B1	F3<67>
Ø1FB230	04	CB	40	Et	81	60	A9 <b8></b8>
0200:09	9D	42	03	49	88	9D	44(52)
0208:03	A9	00	90	45	03	90	49<7A>
0210:03	A9	01	90	48	63	A9	Ø1<7C>
0218:85 0220:00	11 20	4C F9	56 81	E4	85 87	88 FØ	A2 <eb></eb>
0228:A2	20	29 68	68	81 40	CØ	81 82	30<19>
0239:48	A9		20	17		68	THE STATE OF THE S
0240:88		20	00		62		
Ø248:81	8B	00 C9	99	F2	ØF	95	88<66>
0250198	48	A5	88	20	17	87	68(11)
0258:A8	CB	4C	42	82	60	A9	9B<45>
0260:4C	17	82	84	85	CS	90	70.06>
0268:13	A2	20	A9	ØĊ	90	42	03(39)
0270:20	56	€4	10	23	4C	20	82<36>
0278:A9	22	85	87	A5	87	95	95 <bf></bf>
0280: A9	90	65	87	4C	89	82	FD <dc></dc>
Ø288: 45	52	52	4F	52	20	98	A9(A3)
0290:81	05	88	49	82	85	80	20(F1)
0278:40	82	A5	85	85	83	A9	@8<93>
02A0:85	84	20	Di	81	4C	A6	82<65>
ØZAB: 20	28	24	98	A9	AZ	85	88(16)
0290:A9	82	85	BC	20	40	B2	20<38>
ØZBB: AØ	81	A9	29	20	17	82	20<14>
02C0:5B	82	A5	95	85	87	4C	6B<7F>
Ø2C8:84	A9	00	85	BB	85	BC	AA< 74>
02D0:A5	9F	C5	BB	00	A5	90	C5 <d7></d7>
02D8:80	00	68	85	9D	68	25	9D<7B>
02E0:48	28	DØ	05	AØ	05	4C	50 <e1></e1>
Ø2E8:82	98	81	89	€6	87	DID	02<2F>
02F0:E6	8A	20	9C	8E	E.6	BB	DØ <fb></fb>

02F8:02 E6 8C A5 8B C5 8F Ø8<093 0300:A5 BC C5 90 08 48 85 90<25> 0308:68 90 48 28 pa Da AB(49) 25 C9 20(98) 0310:A2 80 A0 00 B1 61 2318: DZ 30 83 CA EØ ØD 20 60<20> AD 89 86 FØ 99 Ø328:18 A9 36 45 80 85 81 A9(35) 0330:05 69 00 83 82 60 E6 80(48) 22 83 20(80) 0338: D0 81 4C 02 É6 **B**1 Ø340: ØA 83 AØ. 60 81 **C9** 20(71) 0348:00 03 20 3.0 63 40 9A B3<DA> 9359: A9 99 BD 44 02 DB A9 19(6C) 259:80 D9 02 A9 03 BD DA @2<A9> 0360:A9 85 @C A9 85 85 OD<OF> R6 0368:A5 10 Ø9 8Ø 85 10 8D ØE<7B> 0370:D2 93 BD C6 02 A5 94(D7) A5 8D C8 02<CB> 0378:80 C5 02 A9 00 0380: AD 88 C9 **B4** @E AD(13) 86 DØ 90 27 AD 9AKAB: 0388:91 86 C9 80 0390:86 C9 E 1 FØ 03 4C 71 E4<33> 53 DØ F6 18KCAS 0398: AD A9 86 C9 69 BD A9(46) 03AØ: A5 58 50 AD 86 A5<49> 03A8: 00 65 59 BD AE 86 A9 Ø380:80 300 **M**2 A9 RA SD 31 02< DA> 53(50) 52 A9 03B8: A9 @1 85 27 85 03C0: 49 85< 9B> 85 4D 60 49 00 99 03CB: 97 85 99 85 9E 85 9F DB<CA> C9 53 93 C9 BEDR: AD 0.9 86 53 FR 93 ACCARD 0308:77 FØ 04<0B> E4 A5 02 03E0:C9 Ø8 DØ ØA A5 94 **C9** 02KF27 03E8:F0 0C C9 8A F8 88 A9 82<96> @D<660 0.5FBx 85 93 49 ØA 85 94 A5 BSFB: C9 E Ø 18 A5 80 84(46) 85 0400:04 8D B5 00(25) A5 QD. 84 A2 0408: B5 80 9D 86 84 ES EB BICDE FF 85 A9 08 A2 0410:D0 F6 **ØØく52**2 E4<D4> 90 42 83 28 56 0418:A9 0C 0420:10 03 4C 20 82 85<73> A9 80 @SKAED 0429:6A A2 MM A9 M3 90 42 0430:A9 ØC. A9 9D<1F> 9D 4A 03 00 34 0438:48 Ø3 4C 84 45 3A 00<7D> 9D<81> 0440:A9 37 9D 44 83 A9 84 2448:45 83 20 56 E4 10 03 0450:20 82 A2 20 A9 OC. 90 42<F7 0458:03 20 56 E4 10 03 40 20(42) 84A8-82 A9 80 85 87 A9 01 R5<113 88 0468:55 A9 85 A9 02 85<FB> 56 0470:54 26 4A 83 A9 FF 85 08<FF 0478:A9 01 85 09 A9 01 SS(SE) 85 2482:A9 00 BD 44 02 A9 22 BD(A5) Ø488: 2F 02 AS 87 85 95 A9 @@<BF> 0490:85 87 20 58 82 4C 99 84<99> 0499:57 45 AC AC 20 3F 98 A9<DA> 04A0:92 A9 84 85 BC 20(64) 85 BB 24AB: 48 82 A5 95 85 87 20 58(35) 8D FC 02 0480:82 A9 FF 40 28<852 Ø488:87 88 40 15 720 22 22 00<5D) 0400:00 00 92 22<C8> 20 26 20 88 04CB: 00 00 80 08K D83 28 2123 00 00 04D0:00 00 00 0101 0103 00 00 MAK DB> 04DB:00 202 22 20 98 98 00 88KE8> 00(EB) 04E0:00 26 88 88 89 82 99 04EB:00 00 00 20 (202) 00.00 ARKEA) 04F0:00 00 00 00 00 00 00 20<FB> 04F B: 00 90 00 88<88> 00 20 20 00 00<0A) 0500:00 00 00 20 00 90 0508:00 00 00 00 00 20 20 Ø8K 125 8510:98 PG 99 00 00 22 22 20K(1A) 05:8:00 00 90 98 00 00 80 00(22) 88 88 88 88 00 00<2A> 0520:00 80 Ø528: ØØ (345) 100 2362 68 88 88 BB(32) 00 98 0530:00 70 22 20 PB<3A> 96 0538:00 88 60 98 22 22 00(42) 0540:00 22 20 0101 0.05 98 99 PROK 4AD 0548:00 22 22 88 88 00(52) 22 22 98 00(5A) 9550: 28 88 66 20 99 99 0558:00 00 00 00 00(62) 00 00 00 0560:00 22 22 210 88 88 88 00(AA) Ø568: ØØ 00 00 00<723 00 00 99 99 20<7A 0570:00 00 00 99 20 00 0578:00 00 22 ମନ ବମ ମନ ବନ @B(B2) 88 22 0580:00 00 20 99 99 00(8A) 0589:00 90 00 98 00<923 89 0590:00 00 (2)(2) 00 00 20 00 ORK PAS 0598:00 മമ 88 (242) O/A 120 120 88(A2) 05A0:00 88 88 20 OD (AA) 88 00 20 05A8:00 00 00 20 20 00 88 (B2) 05580: 00 Q1Q1 (240) (2)(2) 90 OLOI CHOI MACHA A9(B5) 05BB:00 20 00 2505 20 560 86 A9 D9<A6> 0500:00 BD B3 84 29 BD 05CB: 02 A9 05 AD. DA 92 A9 92(63) 05D0:80 36 02 A9 C0 8D 37 824.28> 0508: A9 88 85 6A A2 20 A9 **8**C<39> 20 03(51) 05E0:90 42 03 56 10 85E8: 4C 2C 82 A9 00 85 87 AD(50)

MSF0: 1F 00 5D AZ 00<BD> 05F8:BD B6 B4 95 80 EB EØ 81<10> 2422 : 02 F6 69 02 85 52 A9 MMKE30 02 AD BE DØ 83<6C> MAM8:80 44 FB 0610:60 ØA 00 90<56> 0618:42 903 20 56 E4 12 03 40(20) 88 A9 42<05> MA20: 2C 82 A2 23 9D A9 9D ØØ<CA> 2C 40 03 0629:03 0630:9D 4B 03 4C 33 3AKEA) 86 0638:00 A9 90 44 03 30 A9 86(80) 20 56 0640:9D 45 03 E 4 10 03<AB> **8648:40** 2C 82 AD B3 84 DØ CØ<6D> FA P4 (BB) 0650:6C SF 4C BF B3 6C 2658: B4 80 80 80 80 82 BØK 993 0668:80 80 8D 8D 80 80 AB A1(D9) 0668: B0 82 B9 BD AD AF AE 80(50) 90 80 8D ED 8D 80<2A> 0670:92 BE 067B:80 80<7B> 88 82 80 80 80 82 88 0690:80 88 80 89 88 88 E3(ES) 91 Ø688: 99 99 98 95 80 E2 F9(C1) ED E1 F3 88<86> 0690:80 84 EB EF E3 069B:A6 E9 F2 ED<72> F3 EB E5 06A0:E1 EE EE 80 80 88 BØ 88(09) 26AG: 82 130 60 20 70 705 47 53KE40 00 00 02 **8680:86 82** 42 02 02(96) 82 82 02 02 02 B2<C4> 8698:02 22 0600:02 82 02 02 02 02 02 **82**<000 06CB: 02 02 85(26) 41 A5 86 A9 EE. 92 06D0:A0 4C CØ A9 646 85 ARKEC> C7 BD 36 82 A9 8D(48) 06D8: A9 86 02 9B A2 36<E5> 06E0:37 A9 99 9D **MAFR-85** E8 E0 BR DO ES 49 MMKD45 90<080 A2 06F0:8D 44 02 20 A7 25 36 44 06F8:42 03 A9 9D 03 A9(1F) 0700:B5 9D 45 03 A9 75 9D ARC BRO P788:03 A9 80 9D 49 83 28 56<86> 8710:E4 C8<68> 10 A9 61 55 18 85 AD 82(AB) Ø718:88 00 85 88 4C 50 0720:20 4A 83 68 68 A9 FD 26\5E> 84 CE <42> Ø728:17 82 4C &B 68 20 0730:86 49 00 85 80 20 0738:20 39 83 A2 00 A1 81 C9(8B) 0740:51 DØ 08 A9 00 SD B3 84(92) 0748:4C 9E 85 **C9** 20 DØ 03 4C(Ø9) 0750:EC 80 C9 4C DØ 07 A9 FF 82 W75B: 85 9R 40 51 BF **C9** 55 DANK STE 9B C9 0750:03 4C 43 07<FB> 5A DØ 0768:A9 00 85 9B 4C 51 OF C9(B0) 0770:41 00 62 4C 0A 98 C9 45(1A) DOZZE: DO DE 40 CE 98 09 44 D0<000 4C 03KE5> 0790:03 11 94 C9 **7**B DØ 8788:4C 68 84 C9 2B DØ 4C<3E> C9 52 0790: DØ BE AF DB 83 45 AFCR15 D798:88 **C9** DO 0.3 **B4** 4C BB(7C) M7AM: C9 37 00 03 40 88 E9<310 07A8:40 00 03 4C 3E 89 58(48) 69 0750: DO Ø3 4C DE 89 C9 56 DØ<023 07BB: 03 89 Ç9 4C FE 46 DØ Ø3<A5> C9 54 87C8: 4C 24 SA DØ 03 4CKEF) **C9** 3F 0708:54 84 48 00 0.3 4C 90(97) 07D0:BA C9 DØ 03 8D<8D> 4C 00 4C Ø708: C9 48 DØ 93 29 C9(31) 0700:38 D0 0B 49 G1 BD FLS. SACKEY 0708:40 BE 85 C9 47 DØ 03 40(14) 07F0:45 BD C9 4E DØ 4C 03 D6<4B> FØ 107F 8: 8D 09 3E 25 AB 86 4C(E4) Ø800:5D 82 20 Ø808:42 00 A1 30 83 20 39 83< 90> 81 20 ASKAR> 1.5 817 0810:85 FØ ØC A2 99 AL 81 C9< BB> FØ 0018:2A 84 C9 410 DID 20(BF) BE 39 83 A2 BB201A4 90 20 (9/9) AI(FD) 30 Ø828: B1 **C**7 FØ 05 AR 0.7 40(72) 0030:50 30 82 82 20 20 39 83(EQ) 0858: A2 00 A1 81 29 15 82 A5<3B) 0840185 D0 EA 20 30 03 AZ 08K94> Ø848: A1 15 81 20 80 A5 85 DØ<875 0850: DC 4C 4B 8F 20 38 23(70) 0858:39 83 A2 00 A1 81 **C9** 8868: FB 25 AB 89 4C 5D 82 20<43> 9848:30 83 A2 20 A9 ØC. 42(19) 9D 20 @B70:03 56 E4 10 **23** 4C 20(62) A9 03 0078:82 A2 28 9D 42 Ø3<B9> Ø888: A9 49 03 A5 90418> 20 90 81 0888:44 03 **A5** 82 90 45 A9(E9) E4 BROD: DR 90 20 03 20 56 104777 20 48 A9 0898: 1A 98 A2 36 90<07> 08A0:42 23 20 56 10 03 4C(AD) ØØ 85 08A8: 2C 82 49 87 68 A8(3F) 78899: 4C 20 82 A9 M1 R5 R7 40(35) 8889:6B 84 20 30 83 28 39 83<DØ> 08C0:20 7E 81 A5 83 80 0808:A5 94 8D 08 03 70 39 83(FC)

Listing zu »Happy-Mon«. Bitte mit AMPEL eingeben.

```
08D0:20 7E 81
               A5 83 8D
                             03(14)
2808: A3 84 8D 25 23 A5 83
                             85(34)
08E 0:89
        A5
            84
               85
                  BA
                      20
                          90
                             RE(A9)
            20
Ø8E8:E6
                9C
                   BE.
                      A9
                          52
                             80<60>
               53
08F8:82
        93
            20
                   E4
                      CØ
                             10(29)
RRER: 03
        40
            20
               82
                   AC.
                      68
                          R4
                             20くんて5
        83
               39
                   83
0990:30
                             81 (AC)
            20
                      20
                          7E
0908: A5
        83
            8D
               @A
                   03
                      A5
                          84
                             BD< 20>
0910-0B
        0.3
            20
               79
                  83
                      20
                          7E.
                             81(12)
            8D
                      A5 84
0918:A5
        83
               214
                  23
                             BD(BB)
0920:05
        83
                  85
                      89
                         A5
                             84(4F)
            A5
               83
0928:85
        BA
                             20<44>
            20
                90
                   BE
                          88
0930:9C
        BE
            A9
               57
                   80 02 03
                             20<00>
        E4
0938:53
            CØ
               88
                      83
                         4C
                             20(34)
                  10
0940:82
               84
                   29
                      30
                          83
                             20KC2>
2948:39
        83
            20
               7E
                  81
                      A5 83
                             85<9E>
                             85<2B>
        A5
            64
Ø950:8D
               85
                  8E
                      A5
                         83
0958:89
         AS
            84
               85
                   BA
                      28
                          9C
                             BE < 2A>
                             83(66)
0940:20
        39
            83
               20
                  7E
                      81 A5
0968:85
               84 85
                         20
        89
            A5
                      BA
                             90KEA>
0970: BE
         20
            39
               83
                          81
                             AS(Ca)
                  20
                      7E
0978:83
        85
            BF
               A5 84 85
                          90
                             A9KEA>
0980:00
        85
            88 85 8C
                      AA
                         A5
                             8R(7A)
               A5
0788:C5
                  8C
                      Ç5
        BE
            89
                          90
                             @8<17>
                      9D
                         48
                             28<94>
0990:68 85
            9D
               48
                  25
0998: DØ Ø3
            40
               DE
                  82 A1 BD
                             81(EB)
09A0:89
        E6 BD
               DØ 82
                      E6
                         BE
                             A5<F7>
09A8:89
        48
            A5
               84
                  48 A5 8D
                             85<FD>
09R0: 89
        A5 8E
               85 8A
                      20
                             8E<14>
                          90
0988:68 85 8A
               68 85 89 E6
                             89<97>
09C0: D0
        02
               BA
                             E6<03>
            E.6
                  20
                      9C
                         BE
29C8:88
        DØ
            02
               E6
                  BC A5
                         88
                             C5<83>
9900-96
        88 AS 8C CS 98 88
                             68<6C>
                  9D
                      48
                         28
@9D8:85
        9D
            68
               25
                             DØ<89>
09E0:BC
            6B
               64
                   20
                      30
                          83
                             28(88)
B9EB: 39
        83
            20
               7E
                   61
                      A5 83
                             R5(FF)
89F8:89
        A5 84
               RS RA
                      20
                         30
                             83(01)
        8F
            81
               A2
                  88
                      A5
                         83
09F8:20
                             81(46)
            óΒ
0A00:89
        4C
               84
                  20
                      30 83
                             28<78>
PLADER - TO
        83
            20
               7F
                  R1
                      45 83
                             R5(59)
        A5 84
0A10:89
               85 8A
                      20
                             B3<A3>
                          39
ØA18:20
        Æ
            81
               AB
                  72
                      A5
                          83
                             91<082
0A20:89
        ARI.
            01
               A5 84
                      91 89
                             ACC (E)
MAZE: 68 84
               38 B3
                             83<6B>
            20
                      28
                         39
                      85 89
0A30:20
        Æ
            81
               A5 83
                             A5<3C>
ØA38:84
        85
            BA
               28
                  9C
                      8E
                          20
                             CATINET
               R1 A5 83 85
                             BE<1E>
BA49: 83
        28
            Æ
        84
            85
                   20
                             20<34>
2048: A5
               90
                      39 83
0A50:8F
        81
            A4
               83
                  20
                      C3 82
                             40<05>
0AS8: 68 84
            22
               30 83 20 39
                             83(18)
        7E B1 A5 B3 B5 B9
0660:20
                             A5< ØC>
2868:84
         85 BA
               28
                   9C
                      8E
                          20
                             39<26>
0A70:83
        A2 88
               A1 81
                      C9
                          22
                             F8<50>
0A 78: 0.5
        4C 5C
               AB.
                  20
                      30 83
                             A1KRED
                   12
0A80:61
        C9
            22
               FØ
                      C9
                          9B
                             FB(A5)
        81 89
               E& 89 DØ
0A88: 0E
                          82
                             E6<C8>
                             A5<3C>
4C<9F>
0099:90
        26
            9C
               BE.
                   4C
                      76
                         8A
        85 97
0A98:89
               A5 8A 85
                          98
00 : NAA0
        84 A9
               FF
                   85
                      9C
                          28
                             30K SE 2
0AA8:83
        20
            39 83 A2
                      88 AL
                             81<66)
0A80: C9
        42 F0 0A C9
                      57
                         DØ
                             Ø3<A4>
9AB8: 4C
        48 8C
               4C
                  50
                             38<80>
                      88
                         20
               83
                   20
0AC0:B3
        20
            39
                      7E
                          81
                             ASKEC)
0AC8:83 85 89 A5 84 85 8A
                             45<09>
0AD0:83
        85
            90
               A5
                  84
                      85
                          9B
                             28(6C)
0AD8:90
         BE
            28
               39
                   83
                      20
                          8F
                             81<CA
ØAE.0: A5 83 85
               71
                   20
                      39 83
                             A2<6E>
                   4E DØ Ø4
                             A9(48)
00 BEAR
        A1 81
               C9
        85
                   48
0AF0:00
            90
               20
                      BE A2
                             00<D2>
0AF8: A1
        89 C5
               91
                   De
                             3EKA1>
                      2B A9
8B88:28
        17 B2
               A5 89
                      85 83
                             A5(88)
               26 B2 81
                             3A(37)
0508:8A
        85 84
                         A9
0910:20
         1.7
            82
               A5
                  91
                      85
                         83
                             28(32)
MR18- AM
        B1
            20
               58 82
                      A5
                          90
                             FREDRO
0B20:0B
        20 6D 8B F0 03 4C
                             AB(ED)
               DØ Ø2 E6 BA
                             28(A2)
0B28:84
        E6 89
0B-30: AB
         BE
            A5
               B9
                   C5
                      9A
                          68
                             ASKAED
        C5
BB38:8A
            98 28 48 85
                          90
                             6B(2E)
@B4@: 25
        9D 48
               28 DØ AD
                         20
                             58(65)
@B48: 82
        4C
            50
               BB
                  57
                      68
                          6F
                             6C<85>
0B50:65
        28
            6D
               65 60
                      6F
                          72
                             79<15>
095B - 20
        63 68
               65
                      6B
                         65
                             64(51)
                  63
8860:2E
        9B A9
               46
                  85 8B A9
                             88<9C>
               40
                             82(AF)
BB48:85
        80
            20
                  82
                      20
                         58
        6B
            84
               A5
                   87
                      85
                          95
                             A9<44>
0879:00 85 37
                             9D(E2)
               4C 8C 8B 9C
PRRPP-43
        AF
            6E
               74
                  69
                          75
                             65<1B>
                      6E
ØB88: 20
        28 59
               2F
                   4E
                      29
                          3F
                             59<98>
         9B A9
                  85
                         49
                             88<2F>
0890:1E
               78
                      8B
ØB98:85
        80
            20
                40
                   82
                      A9
                             8D<9A>
                             42<63>
08A0:FC 02 A2 10 A9
                      ØC.
                         90
ØBA8: Ø3
        20 56 F4
                  100 003
                          4C
                             20(AB)
OBBO: R2
        AD
            30
               @2 C9
                      AS FO
                             97<E8>
                   4C
OBB8: A9 FF
            85
               89
                      86
                         88
                             A2<14>
MBC0: 10 A9 83
               90
                   42
                      93
                         A9
                             84<A60
```

```
09CB:9D 4A 03 A9 00
                      9D
                          48
                             Ø3<A6>
8808:4C DØ 88 48 3A 00 A9
                             CD< 98>
GROR: 9D
         44
            P3 A9
                  SB
                      9D
                          45
                             D3<57>
        56
               10
                             82<8D>
            E4
                      40
                          20
0BE0: 20
                   93
         07
               42
                          96
                             9D<53>
OBEB: A9
            9D
                   83
                      A9
PREP- 44
        PCS.
            49
               88
                   90
                      45
                          83
                             90K B70
                   9D
                          83
                             20(00)
0BF8:49
        83
            89
                      48
               01
0C00:56
                      9C
                             17<0E>
         E4
            10
               18
                  49
                          26
@C28:82
        49
            FD
               20
                   17
                      82
                          CØ
                             88(F7)
               75 88 A5
                             85<48>
OC10:00 03
            4C
                          95
               84
                             OC(27)
BC18:87
        4C
            6B
                   A2
                      18
        42
                   56
0C20:9D
            03
               20
                      E4
                          18
                             83<E2>
                   9C
ac28:40
        20
            B2
               49
                      20
                          17
                             82(49)
                      94
                          Ç9
8C38: A5
         95
                   AS
                             4EK&CD
            85
               87
0C38:D0
        83
            C9
               88
                      C9
                          59
                             FREDAD
                   60
0C40:0C
        C9
            98 FØ
                  68
                      A9
                          FD
                             28<0F>
QC48:17
        82
                             30KE0>
            4C
               ØC.
                   80
                      60
                          20
0C50: 83
            39
                             A5<A1>
        20
               83
                   20
                       ZE.
                          81
0C58:83 85 89 A5 84 85
                          88
                             A5<26>
                             28KE92
0C60:83 85
            98
               A5 B4 B5
                          98
0C68: 9C
        BE
               39
            20
                   83
                             81(24)
                      20
                          7E.
0C70:A5
        62
            85
               91
                   A5 84
                          85
                             92<64>
0C78: 29
        39
            83 A2 00 A1
                          81
                             C9KCA>
        DØ
ØC8Ø: 4E
            214
               A9
                   88
                      85
                             20(38)
                          9C
ØC88: 4B
        BE
            A5
               10
                   29
                      80
                          85
                             10KFE>
0C90:8D 0E
            D2
               20 AB BE
                          A2
                             88<82>
8C98:A1 89
               91 DØ
                             89(89)
            C5
                      45
                          E<sub>6</sub>
8CA8: D8
        02
            E6 8A
                  A2
                      88
                             89<6D>
                          A1
            DØ
@CA8: C5
        92
               3D A9
                      3E
                             17(EA)
                         8A 85<9F>20 17<76>
0CP0:02 A5 89
               85 B3 A5
        20
                  A9
GC88: 64
                      34
            82 81
                             B1(A2)
0CC0:82
         A5
            91
               85
                   83
                      20
RCCB: AR Rt
            20
               31 82 A5
                          92
                             85<5C>
0CD0:83
        20
            AØ 81 28
                      58 82
                             A5<46>
9CD8: 9C
        FØ
            ØE.
               28
                  6D
                      88
                          FØ
                             09KF6>
8CE8:4C
        áΒ
            84
               E6 89 D8
                          82
                             E6KC1>
OCER- SA
        28
            AB
               BE
                   AS 89
                          CS
                             PA(FA)
0CF 0:08 A5
                   98
                          68
            BA C5
                      ØB
                             85<43>
OCF8:90
        68
            25
               9D
                   48
                      26
                          100
                             23KE6>
9D90:4C
        400
            BB
               4C B1 8C
                          20
                             TOZOGS
                          81
2028:83
        29
            39
               83
                             A9(F2)
                   20
                      7E
@D10:24
        28
            17
               82
                   20
                      B2
                          81
                             A8<75>
0D18:01
         20
            31
               82 A9
                      3D
                          20
                             17(82)
0D20-82 A0
            Ø1 1
               29 31 82
                          28 D1<26>
                             A5<71>
2028:81
        20
            58
               82
                   4C
                          84
                      6B
8D38:94
        C9
            0A D0 0B A9
                          82
                             85<ØC>
@D38:94
        69
            28 85 93 40
                          AB
                             84KBE>
2040:A9 0A
               94
                  49
                             93<F6>
            65
                      02
                          85
@D48:4C
        6B
            84
               20
                   30
                      83
                          20
                             39< 7A>
0050:83 A2
            20
               A1 B1 C9
                          28
                             FR(99)
RD58: 28 20
            7E B1 A5 B3
                             97<8C>
                          85
0060:A5 84
               98 AS
                      87
                          85
                             95< BLD
            85
8048:A7 00
            85 87
                   20
                      58 82
                             20(D7)
MD70:AD BB
            FR
               21
                   40
                      7F
                          RD
                             4EKE50
               65 78 65
                             75<46>
DD 78: 6F
                          63
         74
            20
8D88:74
         65 64
               98 A9 71
                             89<03>
                          85
               8C 20 40 82
6B 84 4C A1
0088:A9 80
            85 80
                             20(0B)
                             BD(AF)
            4C
2D90:58 82
MD98:52
         75
            6E
               6E
                   69 6E
                          67
                             20(B9)
80A0:66
           6F 6D 20 24 9B
8B A9 9D 85 8C
         72
                             A9478>
BDA8: 97 B5
                             28(42)
0050:48
        82
               97
                             98<84>
            A5
                   65
                      83
                          AS
                             BD<95>
            20
                          BF
@DB8:85 84
               B2
                   81
                       4C
@DC@: 28 2E
            2F
               ZE.
                   98 A9 BA
                             B5(BF)
BDCB: BB A9
               85 BC
                             82(18)
            80
                      20
                          48
0DD0:20
        58
            82
               20
                   D3
                      BD
                          4C
                             6B(D1)
0DD8:84 6C
            97
               88 A2 28 A9
                             ØC<55>
            03 20 56 £4
                          18
                             93<A1>
ADE 2: 9D
        42
@DEB: 4C
            82
               A9
                      85
                          87
                             4C<68>
        2C
                   89
                          39
2DF 0: 6B B4
               38 83 20
                             83<20>
            20
                          97
@DF8:20
        7E
            B1 A5 B3
                      85
                             A5<BA>
0E00:84 85
            98
               4C
                   6B B4
                             D4<CA>
                          ØF
0E 98: 00
        DE
            A9
               80
                   85 85
                          AD
                             31<EF)
                             85<7E>
0E10:02
        C9
            86 FØ 07
                      49
                          (2)(2)
@£18:81
        6C BA
               00 A5 89
                          CD
ØE20:8E
         88
            45
               88
                   CD
                      81
                          8E
                             BB<E1>
                             28<FB>
            9D
                      90
@£28:68
         85
               68
                   25
                          48
Ø£ 30:90
        02 D0 01
                   60 A5
                          29
                             CD<@C>
ØE38:02 BE
            08 A5 8A CD
                          03
                             8E<59>
0E40:08 68
            85
               90 68
                      25
                          9D
                             48<27>
ØE 48:28 90
            81
               60
                   A9
                      04
                          85
                             85(5D)
0E50:60
        28
            BA SE
                   A5 85
                          FR
                             49<FD>
0E58:28
        58
            82 A9 B1 65
                          55
                             A9<46>
@E.68: 01 85
            52
               4C 85
                          20
                             2DKF2)
                      BE.
BE.68: 3E
         20
            24
               44
                   34
                      31
                          30
                             20K5B)
        20
            24
0E70:2D
                44
                   57
                       46
                          46
                             20<63>
8E78:63 61
            6E
                6E,
                   6F
                      74
                             62(0E)
                          20
0E80:65 20
            61
               63
                   63 65
                          73
                             73(A4)
ØEB8:65 64
            9B A9 68
                      85
                          88
                             A9< 2D>
0E90:8E 85
            BC
               20
                   40
                      82
                          28
                             58<?8>
ØE98:82 A9
               85
                             BSKA82
            00
                   89
                      A9
                          D8
REAR-RA AR
            20
               014
                   8E
                      A5
                          85
                             ERK075
@EAB: 07
        6B
            68 AØ
                      4C
                   04
                          50
                             82<F7>
ØEBØ: 60
        A5
            ARI
               FØ
                   26
                      68
                             A5<4F>
                          68
ØEB8:87 85
            95
               A9
                             A9(6B)
```

17 82 A5 95 85 B7<75) GEC8: AD AA 86 C9 86 F0 03 4C<48> @FD@:77 F4 40. **AR 84** 60 20. 30(67) 45<1C> @ED8:83 20 39 83 20 7E BI 85 84 40<28> 9EE9:83 9E. AS. 85 9F DEFR: AR 84 69 710 85 80 20 724370 20(00) 0EF0:83 39 83 20 83 20 30 **BEF8:39** 82 20 A4 60 83 85<8A> 0F00:89 45 84 85 FLA 20 90 REC7AS @F @8: A5 89 85 **A5** BA 98<03> 97 85 @F10:20 39 20 30 82 00<87> 83 AIKEF) @F18:98 48 39 83 A2 22 20 0F20:81 28 15 80 AS 85 DØ 11(51) @F 28: 20 91<56> 78 80 AB A5 83 68 FØ CB CO 28 12<40> ØF30:97 85 4C 8F38:8F 68 AB 84 8F 18 45 97(9A) @F40:65 98<24> A9 85 97 8F 20 65 AS 44 QF 48:85 98 20 20 83<FF> 28 @F50:60 20 E4 BE 4C 87 20<BD> 2F58:32 83 81(97) 20 39 83 7E 85 2F68: AS 83 87 AS. 84 85 BAKA72 @F 68: 20 90 BE A5 89 97 ASKEC> 85 RF 78:8A 85 98 A9 FF 85 9C 20<15> 83 C9 4E (FA) @F78:39 A2 20 A1 81 @F80:D@ 64 A9 20 B6<192 66 65 90 97 DE 88: 8F 4C 88 BF A5 85 89(63) 98 A5<68> @F90:A5 BA 4B 85 20 8E 97 A9KFA> DF98:89 85 **A5** 8A 85 78 20 17 82 98 85 84 OF AD - RE Δ5 97 85 B3KE62 **BFA8: A5** 20 B2 A9<70> 81 0FB0:3A 20 17 82 AØ 80 B1 97<F4> ØFB8:85 83 98 48 20 AØ 81 ARK 835 RECO: 01 28 31 82 AB A8 CS CO<D3> DØ 21 0FC8:08 EB A5 9B FØ AB(FD) 97 94 48 A949A3 0FD0:00 81 85 98 **ØFD8:18** 28 17 R2 A5 9A PQ 98(C3) 84 A9 REER: DR 28 17(88) 20 85 68 C8 DØ GFEB:82 48 AS CO 08 REFR: AZ 88 E6 97 DØ Ø2 E.6 98(71) B245E> REFERER EN NR DN ES 20 58 1000:20 AB BE A5 90 FØ 20<9D> 25 1008:4D 8B D0 01 4C 84<38> 60 6B 1019:20 30 83 20 39 83 20 7F<4C> R4 B5<55> 1018:81 A5 B3 85 89 45 97<5A> 1020:BA 20 9C BE **A5** 87 85 1028: A5 88 85 98 49 97 85 52(05) 58 82 A9 A9KCB> 1939:28 07 85 52 1038:01 85 55 A5 87 85 95 A9(37) 20 98 1040:00 85 87 A9 3E 17 82<2F> 85 83 A5 84<63> 1048: A5 97 85 B2 1050:20 81 4C 54 92 3A 1F (@9) 1058:1E 98 49 50 85 BB A9 90(45) 1969:85 8C 29 40 B2 **A5** 95 B5(BB) 1068:87 CE 20 86 **A9** 01 85 52<9A 1070:A9 01 85 55 AP 00 85 80<7B> 1878:20 22 B3 A2 22 A1 81 C9(FA) 1080:98 DO 15 4C 84 98 FD 10(42) 1089:90 9B BB 90<F2> A9 80 85 97 1090:85 BC 20 40 82 4C 2D 90(85) 1098:20 39 83 20 7B 8/2 A7 00(88) 89 28 1000:85 22 **B3** 20 10 BE (93) 90 10AB:4C 2D 49 8D D2<CA> 99 08 03 BD ØF 10P0: A7 D2 A9 78 8D<3D> 1068:00 D2 A9 A5 80 Rt **D**2 A9(71) 20 02 1000:03 BD 02 AD 20 DØK 6B> A9 88 10C8:FB 80 22 02 A9 00<02> 30 20<96> 1000:80 01 D2 60 20 83 81 10DB: 39 83 20 7E **A5** 83 B5<FC> 10E0:89 A5 84 BE (99) 85 20 9C BA 10E8:A5 89 85 97 98(29) A5 BA 85 10F0: A9 27 85 52 20 58 82 A9KBC) 10FB:07 52 A9 85 01 85 35 A5<10> 95 A9(12) 1100:87 85 A9 00 85 87 9C 20<36> 1108:FF 85 9E A9 22 85 1110:86 BF 3B A5 97 E'9 ØB B5<4F> 1118:97 A5 98 E9 00 85 98 A9<60> 1120:07 85 55 A9 10 20 82<7D> 1128:A5 95 85 87 201 CE BA. A9< D7> 1130:01 85 52 A7 @1 85 55 A9(7A) 1138:00 85 80 28 39(42) 22 83 20 1140:83 49 20 78 80 1848 85 80<89> 22 1148:20 83 20 10 8F 4C F1<7B> 1150:90 42 52 4B 30 4F 52 41(29) 1159:37 3E 3F 3E 4F 52 41 33(A6) 1160:41 53 48 4C 3F 33 50 50(AE) 1168:30 4F 52 41 34 41 53 4C<4C) 1170:31 SE 3F 4F 41 52 32 41(EC) 1178:53 4C 32 SF. 42 50 40. 42(AB) 1180:4F 52 41 38 3F 5F 4F < 2E > 3F 41 39 1188:52 41 53 4C 3F < D4 > 1190:43 4C 43 30 4F 41 52 36<3F>

Listing zu »Happy-Mon« (Fortsetzung)

1698:80 97 DØ Ø2 E6 98 A1(EE) 39 BS 20 7E 81(D9) EA 1198:3F 3F 3F 4F 52 41 35 41(42) 1418:38 RS 28 85 BE E6 97 D0 02 4C 1429:A5 83 85 89 A5 B4 85 BAKSE) 16A0:97 E6<6F> 3F 4A 53 52 32<6E> 11A0:53 35 A5 A3 D8 13 A5 BD 37 3F 3F 42 49<9D> 1428:20 9C AS 89 85 97 A5<93> 16AB: 98 85(86) 11A8:41 4E 44 1430:0A 85 98 28 39 B3 A9 BB(9C) 1ARR: RX 26 AR S1 A5 SE 85 B3(31) 1180:54 33 41 4E 33 4F<6F) 02<5B> 4C 98<98> 20 31 33 85 A3 A2 1488:28 A@ 81 AB 82 ూ 50 58 39 41<7D> 1438:85 90 49 98 1188:4C DØ 84 FF(B5) 1608:28 A9 24 82(87) 4E 6D 95 1448: A1 81 **C9** 11C0:4E 44 34 52 4F 4C 31 3F<69> 16C8: A5 BD 85 83 A5 BE 84(71) 9C A5 97 85 89 A5 98(BF) 49 54 44 32<69> 1448:65 83 1108:42 32 41 4E 1600: 28 1450:85 8A 28 48 BE AT 3E 28(61) 82 81 49 7C 208 17 82(65) 1100:52 4C 32 3F 42 40 49(82) 85 83 A5 98(02) 39F 3F<72> 82 60 88<77> 41 4E 44 38 3F 1458:17 82 A5 97 16D8: A9 59 20 17 A2 1108:42 02(96) 20 B2 81 A9 3A 28<DE> 16E0: A1 97 65 83 E6 97 1460:85 84 11E0:41 4E 44 39 52 4F 4C 39(E3) 94 A3 D8 88 20 A8<7A> 44<A0> 26 58 R2(8E) 53 4E 1468:17 82 79 16E8: E6 98 A5 11EB: 3F 45 43 30 41 3F 3F 4E 44 35<B7> 1478: 28 AB BE AS 9C DØ DX 78<45> 16F0:81 ARI DIA 20 31 82 20 ADK BBD> 11F@: 36 3F 41 AZ(CA) 1478: AD BB FB CE 4C AB B4 24(48) A9 28 20 17 82 A9 11FB: 52 4E 4C 35 3F 52 54 49(28) 16FB: 95 83 97 85 45 A3 D8<80) 1780:20 17 82 AB 61 4C 図7(84) 1200:30 45 4F 52 37 3F 3F 3F ( DØ2 ) 1488:00 A1 20 20 1488:03 20 AZ. A1 97(18) 1788:97 9B A9 1208:45 4F 52 22 40 53 52 33(95) All 81 20 58 29 85<34> 50 **B**5 48 41 39 45 45 52<8A> 1498:85 98 EA 97 DØ 62 E6 98<5A> 1710:88 A9 97 BE 28 40 82<FE> 91 A9 91 92(CC) 1218:34 4C 53 52 31 395 40 4DC5E> 1498: A9 48 85 85 171B:60 A2 00 A1 97 B5 83 20<50> 32 45 52 32 40 53(B3) 14A0:A0 00 C4 9B FB 23 A2 88<1A> 1720:39 83 EA 97 DØ 62 98<9C> 1220:50 4F E6 32 3F 42 56 43 42 45<4D> 1448:A1 91 **C**9 3F DØ E6 91(65) 1728: A5 A3 DØ Ø8 20 AØ 81 1228:52 1230:4F 52 38 3F 3F 45 4F<1B> 1480:DØ 02 E6 92 CB 4C 9C 94<73> 1730:04 82 20 6D A9(2B) 3F 20 31 95 1238:52 39 4C 53 52 39 3F 43(3A) 1488: 18 A5 91 69 84 85 91 A5<B7> 1738: 2B 20 17 B2 A9 24 20 17<7A> 92 4C 45 97 29(02) 49 4F 20 9C<F6> 1740:82 29 AB 81 1240:4C 30 45 36 3F(FD) 1408:92 69 85 CB 40 52 45 4F 4C 53(27) 14C8:94 DØ(33) 41 1248:3F 3F 52 35 A2 88 A1 91 C9 3F 1748:2C 39 9B Α9 85 A9<303 300 1250:52 35 3F 52 54 53 14D0:44 A5 A3 D8 18 4C DE 94(BE) 1750:97 85 BC 28 48 82 A2<25> 41<00> 68 37 44(83) 29. 29 20 20 20<19> 83 1258:44 43 3F 3F 3F 41 14D8:28 28 20 1758:00 41 97 85 E6 97 90<CA> 33 9B 1260:43 52 4F 52 33 3F 58<4F) 14E0: 3F 3F 3F A9 02 25 BB<313 1760:02 63 98 AS A3 DØ 20(EA) 88 60<98> 1268:4C 41 38 41 44 43 34 52(C2) 14E8:A9 94 85 BC 26 1768:AB AB 29 29 31 40 82 81 04 28(44) 52 1270:4F 31 3F 4A 4D 58 43(15) 14F0:4C FC 94 20 20 28 28 20(83) 1778:6D 95 A9 24 17 62 20(82) 44 43 20 2E A9(F6) 1278:41 32 52 4F 32 32(3E> 42 59 54 45<7C> 20 28 17 14F8: 20 28 1778: AD B1 A9 82 42 56 42 41 44 98 ED B5 28 A9 94(1C) 1280: 3F 53 43(A5) 1500:20 A9 1780:58 20 17 82 68 A1<14> 1288:38 3F 3F 3F 41 44 43 39(2E) 1506:85 8C 28 48 R2 A9 90 85(13) 1788:97 85 83 EA 97 DB 82 E6(18) 4F B1<87> A3 1290:52 52 39 53 45 D1 81 AB 83 1790:98 20 A0 81 A5 3F 49(A2) 1510:84 28 68 DØ. 05<F4> 3F<39> 60 1298:30 41 44 43 36 3F 36 151B: 91 28 15 **C9** 200 1798: AØ 284 29 31 92 95(A9) 20 20 20 AB 44 43 35 52 4F 35<C9> 1520:4C 95 95 C9 01 00 230 4C<5@> 17A8: A9 24 17 82 B1<F8> 12AB: 41 52 B4<585 53 20 17 59 1208s 3F 3F 54 41 37 3F 3F (70) 1528: AZ 95 C9 82 D8 83 40 17AB: A9 82 A9 20(35) 85(29) 1530:95 C9 03 D0 03 4C FS 95KFC> 1780:17 1280:53 59 33 53 54 41 82 60 A2 99 A1 54 33(41) 54 1538:C9 84 D8 83 4£ 19 96 C9(AE) 1788:83 85 9A Éő 97 100 02 1298:53 58 33 3F 44 45 39<23> E6(EQ) 1200:30 3E 54 38 41 38 39-53(3R) 1548:85 DB EICS. 40 42 96 C9 86(83) 1700:98 A5 A3 DØ 98 20 AØ B1<D8> 32 62 40 BD 96 67 1208:54 59 53 54 41 32 53(C3) 1548: DØ C9 D6<68> 17C8: AØ 84 20 31 82 20 6D 93(09) 58 1550:03 4C DB 96 C9 83<2C> 29 1200:54 32 3F 42 43 43 42(CE) 88 D8 17D8: 49 24 17 82 A3 C9(56) 54 39 41 53 13 97 1208:53 38 3F 3F 33 54 (AE) 1558:40 97 C9 89 DB 83 4C(B6) 1708:88 90 17 38 49 86 E3 984733 4C 97 7F<34> **C9** 8A D8 83 E5 1288:59 54 41 39 53 S4KAE> 1560:51 17E0:85 **9A** 38 A5 9A B5<D1> 17EB: 58 41 3F 54 59 41 30 53<F@> 1568: 97 C9 88 D8 83 40 AD 97(36) 17EB183 AS 98 €9 88 85 84 40<89> 18 41 36 54 1570:4C FD 97 A5 A3 DB 85 A9<54> 17F0:F9 97 **A5** 97 65 9A 12F0:54 58 53 38 3F < 89 > 85<A8> 20 53 12F8:3F 54 41 35 3F 3F 40<56> 1578: 2D 28 17 BZ A9 20 17(27) 17FB: 83 A5 98 69 88 85 84 20(05) 88 1588:82 AB 99 BI 91 85 98<2C) 81 60 A2 98 A1 97 85(D6) 1300:44 59 34 4C 44 41 37 4C<A7> 1800:B2 E6 34 4C 44 A5 97 DØ 82 308:44 58 3F 39 33(25) 1568: 48 88 82 68 ABK 4B) 1808:80 44 54 181**0:97 85 BE** 1818:98 A5 A3 1310:4C 41 33 4C 44 58 33(18) 1590:C8 C8 03 D0 EE AB 81 20<FB> E6 97 DB 82 E6<FB> 41 A5 A3 DØ 05 A6<38> DØ 13 1598:31 82 68 A5 80 85<AA> 1318:3F 59 30 4C 44 41(10) 68<55> 41 44<6C> 1320:34 54 58 38 3F 4C 28 31 20 95 1820:83 20 AØ 81 BE 44(C2) 1588: A5 A3 D0 05 A0 06 1580: 82 20 AD 95 A9 41 1828:28 A& 81 A& 1830:28 AD 95 A9 20 31 1328:59 32 4C 44 41 32 40 28 31(A6) 82 82(DF) 20 28 20 1330:58 32 3F 42 43 53 42 4C(D2) 17<50> 17 82<3A> 4C 1338:44 41 38 3F 3E 44 59(57) 1588:82 60 **A2 A1** 85 BD(E9) 1838: A5 BD 85 83 A5 8E 85 84<7E) 40 15C0: E6 DØ 02 E6 1840: 20 B2 91 A9 29 20 28 39 1340:39 44 41 39 4C 44 58(F9) 97 98 A1 97(27) 17 82(09) 98(89> 1348: 41 3F 43 4C 56 38 40 44<BB> 15CB: 85 BE EA 97 DØ 82 E6 1848: 60 20 30 83 83 20<77> 54 1500: A5 A3 D8 13 A5 80 85 83<4E> Bi A5 83 85 89 84<C4> 1350:41 36 53 28 26 3F 4C(21) 1858: 7E 85 20(30) 1358:44 59 35 4C 44 41 33 4C(AB) 15D8: 20 AØ 81 A5 8E 85 83 1858:85 BA 20 90 BE A5 89 83<56> 1360:44 58 355 3F 43 50 58 34<75> 15E0:A0 B1 A0 02 20 31 82 20(85) 1868:97 A5 8A 85 98 4C 6F 98(8F) 50 A9 28 82 29 1368:43 4D 37 3F 3F 58<7C> 15EB: 6D 95 24 17 A5<80> 1868:31 30 30 30 28 ZE 4F<51> 43 1370:59 33 43 4D 50 33 44 45<662 15F0:8D 85 83 A5 8E 85 84 20<76> 1970:52 47 20 24 9B A9 85(99) 1378:43 33 3F 49 4E 59 30 43<00> 15F9: B2 81 AR AZ RR A1 97 85(DB) 1878: BB A9 98 85 IRC: 20 48 ROCKS. 58 34 1600:83 97 De 62 98 A5<03> B3 A5 98 85 1380:4D 44 45 58 38 3F(55) E6 E6 1890: A5 97 85 84<DA> 1388:43 50 59 32 43 4D 50 32(02) 1608:A3 D9 88 20 AB 81 AB 84(92) 1888:20 B2 81 28 58 82 4C 92(72) 82 95 3F 1610:20 31 20 AD Δ9 24(FE) 1890: 98 30 39 31 29 3B 9B(EB) 1390:44 45 43 32 42 4E 45(96) 31 20 98 1398:42 43 40 50 30 3F 3F 3F<03> 1618:20 17 82 AØ 81 68 AZ(BØ) 1898: A9 8B 85 BB A9 85 8C<8B> 13A8:43 4D 50 39 44 45 43 39(28) 1620:00 A1 97 85 83 E6 97 DØ<88> 18A0: 20 48 58 82 49 86<410 82 20 A5 30 1628: 07 E6 98 **A3** 06 88 28(48) 18A8: 85 90 **A9** FF 85 A3 **A9** E9<18> 13A8: 3F 43 40 43 50<64> 1.3BØ: 36 3É 3F 3F 43 AD 58 35(7E) 16301-00 81 00 04 28 31 82 28(AB) 1888:85 A1 49 60.75 R55 A2 20 39(00) C9 45 43 50 1638:6D 95 49 23 58 17 82 A944A> 1888:83 <u>A2</u> 90 Αį 81 4E DØ<42> 35 3F 43 58(89) 1388:44 1300:34 42 43 37 43(35) 1640:24 20 17 82 28 AB 81 68<23> 1800:04 49 85 90 E6 DØK CA> 58 33 1308:50 53 42 43 33 49(16) 1648:A2 88 A1 97 85 RD FA 97(04) 1808:27 EA A2 A5 97 85 89 A5(78) 1650:D0 98 A1 97 BE(27) 1800:98 8A **4B** 43 3F 49 4E 59 38(28) **62** E6 85 85 20 BE A5 A1<68> 13DØ:4E 33 13D8:53 42 43 34 4E 4F 50 30KC20 1658:E6 97 DØ 02 Eá 98 AS A3(31) 1808:85 83 A5 **A2** 85 84 D1<85) 79(88) 13EØ:3F 43 50 58 32 53 42 43(88) 1668: DB 13 A5 BD 85 83 28 AB(25) 18E0:01 AB 82 28 31 82 20 81(25) 28 18EB: 94 28 82 45(B9) 13FR: 32 49 46 43 32 3E 42 45(05) 1668: B1 85 BE 85 83 28 AB 58 AB BIE 82 20 31 28 ΔĐ 95(80) 18F0:9C DØ 02 20 6D 88 CD(F9) 42 53 42 43 28 1670: AB 82 13FB: 51 35 3F < 52> 53 42 43 39 49 4E 43<93> 1678:49 24 20 17 82 A5 8D 85<F6> 18F8:4C 6B 84 00 00 00 00 00(89) 1400:39 3E 53 45 44 38 53 42<8E> 1688: RS 95 BE B5 84 29 **B2** B1(1C) 1988:88 88 88 88 88 88 88 88(32) 3F 2C 28 17 82 49 28(18) 3F 1688: A9 58 1489:43 36 3F 42 43(29) Laenge 6407 Bytes 53 97 43 35 99 VS 98 Listing zu »Happy-Mon« (Schluß)

»Happy-Mon« greift sogar auf einzelne Diskettensektoren zu. Dies ist speziell bei Boot-Disketten nützlich, da sie keine Files enthalten. Mit »R Sektornummer, Bufferanfang« liest man einen Sektor, entweder mit 128 oder 256 Byte Länge, in den Speicher ab »Bufferanfang«. Mit »W Sektornummer, Bufferanfang« läßt sich ein Sektor wieder zurück auf Diskette schreiben.

#### Hexadezimal-Dezimal-Umwandlung

Gibt man beispielsweise >? 100« ein, wird der entspre-

chende Hexadezimalwert ausgegeben. »? \$100« wandelt eine Hexahl in eine Dezimalzahl um.

#### Bildschirmdarstellung

Mit dem Befehl »K« steht normale oder invertierte Bildschirmdarstellung zur Wahl.

#### Maschinensprache-Unterprogramme aufrufen

Mit »G Adresse« kann ein Maschinensprache-Unterprogramm, das mit einer RTS-Anweisung enden muß, aufgerufen werden. Der Programmcounter und dle »+«-Variable

Zur Vereinfachung der Arbeit können Sie jederzeit die Platzhalter »\*« und »+« verwenden. Dabei repräsentiert »\*« den Programmcounter und wird in den Funktionen L, E, Z, A, D, U, F, und T mitgeführt. Ein mit der Funktion »L« gestartetes Listing setzt »L. \*« fort. »+« ist eine vom Benutzer frei definierbare Variable. Den Wert von »\*« ändert man mit »\* Wert«, und von »+« mit »+ Wert«. Die Befehle »\* +« und »+ \*« sind zulässig.

»Happy-Mon« arbeitet sowohl mit DOS-XL als auch mit DOS 2.0 oder DOS 2.5. Die abgedruckte Version läuft allerdings nur mit DOS-Versionen, die nicht ständig im Speicher verbleiben, wie DOS 2.0 oder DOS 2.5. Für Besitzer von DOS-XL oder anderen residenten DOS-Versionen befindet sich auf der Leserservice-Diskette eine zweite Version mit dem Namen »MASTER1.COM«.

#### Besondere Hinweise zu »Happy-Mon«

Geben Sie zunächst das Listing mit AMPEL ein. Ist »Happy-Mon« komplett abgetippt, speichern Sie das File bitte mehrfach. Legen Sie sich auch eine Kople auf einer weiteren Diskette an, die Sie an einem sicheren Platz aufbewahren. Sollte dann Ihre Arbeitskopie versehentlich gelöscht werden, haben Sie immer noch eine Kopie. Sicher ist sicher!

Aufruf von »Happy-Mon«

Sollten Sie die abgedruckte Version verwenden (auf der Leserservice-Diskette finden Sie das File mit dem Namen »MASTER.COM«), müssen Sie entweder mit DOS 2.0 oder DOS 2.5 arbeiten. Um das Programm zu starten, gehen Sie zunächst mit »DOS« ins DOS-Menü. Dann laden Sie den Maschinensprache-Monitor mit der L-Funktion. Kehren Sie anschließend mit B zum Basic zurück, jetzt können Sie beispielsweise ein Basic-Programm eingeben oder von Diskette laden. »Happy-Mon« rufen Sie mit »? USR(32768)« auf. Sie befinden sich anschließend im Eingabeeditor des Maschinensprache-Monitors. Hier können Sie dann alle beschriebenen Funktionen ausprobieren. Mit X gelangen Sie zum DOS zurück und mit Q zum Basic. Sollte sich vor dem Aufruf von »Happy-Mon« ein Basic-Programm im Speicher befunden haben, steht einer Weiterbearbeitung nichts im Wege.

Die zweite »Happy-Mon«-Version ist nur auf Leserservice-Diskette erhältlich. Sie finden Sie mit dem Namen »MASTER1.COM«. Diese Ausführung nutzt die Adressen ab \$2200, also oberhalb von DOS-XL. Somit ist ein Zugriff auf die Adressen \$8000 bis \$9FFF, in dem sich »Happy-Mon« normalerweise befindet, möglich. Außerdem ist diese Version Voraussetzung, wenn Sie die Diskettenversion des MAC/65-Assemblers verwenden. Bevor Sie »Happy-Mon 2« einsetzen, ist noch das File »AUTRUN, A« in »AUTORUN.SYS« umzubenennen. Anschließend können Sie diese Diskette booten. Wenn Sie mit der Diskettenversion des MAC/65-Assemblers arbeiten, schalten Sie den Atari-Computer mit gedrückter OPTION-Taste ein, um das Basic zu desaktivieren. Von der DOS-Befehlszeile aus laden Sie dann anschließend das File »MASTER1.COM« Ansonsten sind die Funktionen in beiden »Happy-Mon«-Versionen identisch.

Wichtige Anmerkung

»Happy-Mon« verwendet dle Zeropage-Adressen 128 bis 160! Diese sollten nicht verändert werden. Sprünge von und zu Programmen, die diese Register benutzen, wie beispielsweise Basic und der MAC/65-Assembler, sind allerdings problemlos. »Happy-Mon« rettet diese Register bei der Initialisierung und stellt sie beim Verlassen (auch bei RESET) wieder her.

(Thomas Fischermann/wb)

### Windows: Nicht nur ein Augenschmaus

Mit der Window-Technik läßt sich eine Benutzerführung realisieren, die bislang nur teuren Personal Computern vorbehalten war.

ört man den Begriff »Window« oder Fenster, so verbindet man diese Begriffe mit einer hervorragenden Benutzerführung. In dieser Hinsicht hat besonders der Atari 520 ST von sich reden gemacht.

Das Programm »Windows« für den Atari 800XL/130XE ist sicher nicht mit GEM, der Benutzeroberfläche des Atari ST, zu vergleichen. Vielmehr soll es dem Atari-Fan zeigen, daß sich eine abgespeckte Version auch auf den »kleinen« Computern realisieren läßt. Mit diesem Programm kann man an einer beliebigen Stelle auf dem Bildschirm ein Fenster einblenden. Der Vorteil dieser »Window-Technik« liegt darin, verschiedenartige Texte oder Grafik und Text nebeneinander bearbeiten zu können. Löscht man das Fenster, erscheint auf dem Bildschirm wieder der ursprüngliche Text.

Der Trick dabei ist relativ einfach. Bevor man ein Fenster öffnet, wird der überblendete Bildschirmausschnitt in einen anderen Speicherbereich »gerettet«. Daraufhin ist dieser Bildschirmausschnitt für andere Arbeiten frei. Benötigt man das Fenster nicht mehr, wird der anfängliche Bildschirmausschnitt wieder in den Bildschirmspeicher geschrieben. Soviel zur Fenstertechnik. Wenden wir uns dem Programm selbst zu.

Tippen Sie zunächst Listing 1 mit dem Prufsummer ab und speichern Sie dann das Programm. Nachdem Sie RUN eingegeben haben, wird ein File mit dem Namen WINDOWS.OBJ auf Diskettenlaufwerk Nummer eins abgelegt. Dieses File enthält dann die eigentlichen Window-Routinen. Um es einfach und schnell von Basic aus laden zu können, verwenden Sie Listing 2 (bitte mit dem Atari-Prüfsummer eingeben). Das Window-Initialisierungsprogramm liest dann die Window-Routinen in den Speicher. Da nun die Betriebssystemerweiterung im Speicher vorliegt, können Sie bedenkenlos mit NEW löschen.

Ein anderer Weg, das Programm »Windows« zu laden, lst von Basic aus »POKE 106,144:DOS« einzugeben. Vom DOS-

PROGRAMM-STECKBRIEF				
Programmname	Windows			
Programmtyp	Utility			
Programmiersprache	Basic und Maschinensprache			
Programmlänge	4974 Byte			
für Computer	800 XL/130 XE			
zusätzliche Hardware	keine			
Eingabehilfe	Prüfsummer und AMPEL			
Bemerkung	Maschinenprogramm mit Basic-Lader			
Leserservice	Diskette (LADE.BAS, WINDOWS.COM, WINDBEIS.BAS)			



So könnte ein typischer, mit Windows versehener Bildschirm aussehen

Menü aus ruft man dann »WINDOWS.OBJ« auf und kehrt wieder ins Basic zurück. Nun noch »?USR(37120)« eintippen, und die Fenster sind abrufbereit. Die Betriebssystemerweiterung benötigt 4 KByte-RAM. Zieht man aber die Vorteile der Fenstertechnik in Betracht, dann ist dieser Speicher bestimmt kein verschwendeter Platz. Die Fenster werden in Grafikstufe 0 übrigens genauso angesprochen, wie andere Geratetreiber auch.

#### Offnen eines Windows für Ein- und Ausgabe

\*POKE 1538,[X-Länge des Windows]: POKE 1539, [Y-Länge des Windows]: POSITION [X-Position], [Y-Position]: OPEN #[Kanalnummer 1–7],12,0, "W "<.

Durch den OPEN-Befehl wird ein (zunächst leeres) Window auf den Bildschirm ausgegeben. Dabei enthalten die Speicherstellen 1538 und 1539 die X- und die Y-Ausdehnung (X-Ausdehnung = 5 bedeutet zum Beispiel, daß im Window 5 Spalten zur Verfügung stehen).

Beispiel: »POKE 1538,10.POKE 1539,8:POSITION 4,2:OPEN #1,12,0, "W: "«

#### Beschreiben eines Windows

»PRINT #[Kanalnummer wie bei OPEN]; "TEXT"«.

Die Ausgabe von Text erfolgt, wie man es von den Grafik-Modi 1 und 2 her kennt. Das ASCII-Zeichen 125 löscht den Inhalt des Windwos, und der windowinterne Cursor wandert dann wieder an die obere linke Ecke des Fensters. Wenn der Cursor dann den unteren Rand des Fensters erreicht hat, wird der Inhalt des Windows nach oben gescrollt. Beispiel: »PRINT #1; "5+7=";5+7«

#### Einlesen einer Textzeile

»INPUT #[Kanalnummer wie bei OPEN]; [Variablenname]«.

Der INPUT-Befehl wurde gegenüber dem üblichen INPUT etwas verbessert. Der Benutzer kann mit dem Cursor die INPUT-Zeile nicht mehr verlassen (zum Ausbessern von Fehleingaben stehen die Pfeiltasten und DELETE-BACK-SPACE zur Verfügung). Außerdem kann man die maximale Länge der Eingabe festlegen, indem man den entsprechenden Wert in die Speicherzelle 844+[Kanalnummer\*16] POKEt. Nach einem INPUT-Befehl steht in dieser Adresse wieder eine Null. Enthält diese Stelle keinen Wert großer Null, dann wird die Eingabelänge von der letzten Spalte des Windows begrenzt. Beispiel: »POKE 860,5:INPUT #1;A\$« oder »INPUT #1;X«.

#### Positionieren des Cursors im Window

»X=[Spalte 0 bis X-Ausdehnung (Inhalt von 1538) des Windows minus 1]:Y=[Zeile 0 bis Y-Ausdehnung (Inhalt von

1539) des Windows minus 1]:POINT #[Kanalnummer wie bei OPEN],X,Y«.

Das nächste Zeichen wird dann an der entsprechenden Stelle ausgegeben (der POINT-Befehl funktioniert im Zusammenhang mit dem Window wie der POSITION-Befehl beim Bildschirm-Editor). Beispiel: »X=4: Y=1: POINT #1,X,Y«

#### Schließen eines Windows

»CLOSE # [Kanalnummer wie bei OPEN]«

Der ursprüngliche Inhalt des vom Window verdeckten Bildschirmteils wird wieder hergestellt. Beispiel: »CLOSE #1«.

#### Schließen eines Windows, ohne den ursprünglichen Bildschirminhalt wieder herzustellen

»POKE 832 +[Kanalnummer\*16],255«. Nun kann der Kanal wieder benutzt werden, obwohl das Window auf dem Bildschirm bleibt. Beispiel: »POKE 848,255«.

Beim Öffnen und Schließen eines Windows (mit CLOSE) verlegt der Window-Handler den Cursor in die obere linke Ecke des Bildschirms.

Selbstverstandlich erkennt der Window-Handler auch eventuelle Fehler (zum Beispiel Cursor außerhalb des Bildschirms). Man erhält dann die gewohnten Fehlermeldungen. Dadurch ist es fast unmöglich, die Window-Routinen durch falsche Werte zum Abstürzen zu bringen.

Wer noch einen Atari 400 oder 800 besitzt, muß in die Window-Initialisierungsroutine (Listing 2) die Zeile »32005 POKE 121,254:POKE 122,254« einfügen. Die INPUT-Routine des Window-Handlers benutzt nämlich diese Adresse (KEYDEF), die bei den XL- und XE-Modellen auf den Anfang der Tastaturbelegungstabelle zeigt, zur Tastaturabfrage. Durch Abändern von Listing 1 (Zeile 105 »OPEN #1,8,0, "C: "«) und Listing 2 (Zeile 32035 »OPEN #1,4,0, "C: "... «) läßt sich die Betriebssystemerweiterung an den Atari-Programmrekorder anpassen.

Listing 3 beinhaltet ein kleines Spiel, mit dem Sie die Window-Betriebssystemerweiterung ausprobieren können. Es handelt sich darum, durch Ziehen von Karten und Addition ihrer Einzelwerte möglichst nahe an die Zahl 23 heranzukommen, sie aber nicht zu überschreiten.

#### Windows - technisch betrachtet

Die zentrale Ein-/Ausgaberoutine des Betriebssystems (CIO) benutzt für die elementaren Ein-/Ausgabeoperationen wie Gerät öffnen, Gerät schließen, Byte lesen, Byte schreiben, Statusabfrage und sonstiges, die sogenannten Geräte-Handler. Die Anfangsadressen dieser Maschinensprach-Unterprogramme werden von der CIO über die Handler-Adressen-Tabelle (HATABS, \$031A bis \$033F) berechnet. Durch Ergänzen dieser Tabelle mit eigenen Handler-Adressen lassen sich beliebig eigene Ein-/Ausgabegeräte definieren. (Dietrich Wagner/wb)

Initialisierungsprogramm
(Window-Handler wird in
HATABS eingetragen) \$9100-\$9118
Window öffnen \$9119-\$92A4
Window beschreiben \$92A5-\$93B3
Sonstiges (Cursor positionieren) \$93B4 \$93DF
Textzeile einlesen \$93E0-\$961F

Zwischenspeicher für den vom Window verdeckten Bild-

schirminhalt Systemvariablen für den \$9800~\$9C99

Window-Handler \$0600-\$062F

Wichtige Adressen des Programms »Window«



```
0000:FF FF 00 91 1F 96 68 20-08
                                         2188:91 A9 00 80 10 66 80 11<FA
                                                                                  0370:A9 FF 8D FC 02 AD FC 02(BC)
0008:F6 95 B0 03 20 08 96 60(67)
                                         01C0:06 A0 01 60 80 12 06 AE(6A)
                                                                                  0378:C9 FF F0 F9 AA 20 23 95<E1>
        91
                                                        00 06 18 65 58(15)
0010:18
           37
              92 DF
                     93
                           92<62>
                                         01C8:01 06 AD
                                                                                              27 DØ Ø3
                                                                                                           7B 95<D3>
                        Α4
                                                                                  0380:8A CY
                                                                                                        4C
              93
                     10
                        96
                           20K3A3
                                                              98
                                                                                  #388: C9
                                                                                              08 83
                                                                                                    40
                                                                                                           95 AB(E2)
0020:BB 92 A5 54 1B 6D 03 06<96>
                                         0108:06 CB 18 69 28 90 01 E8<D2>
                                                                                  0390:81 79 8D 26 06
                                                                                                       C9 7D F8<CF>
0079:90 06
           30 04 C9
                    17
                        30 07(D1)
                                         81E0:88 D0
                                                     F7
                                                        18 6D 19 86
                                                                     99(42)
                                                                                  8398: CE C9
                                                                                              7E DØ Ø3
                                                                                                        4C
                                                                                                           3C
                                                                                                              95(F9)
                                                 E8
                                                     18
                                                        69
                                                                                   BIAD: C7
0030:A0 80 A9
              FF
                  85 20
                        68
                           A5< 9A>
                                         @1E8: @1
                                                           91
                                                              98
                                                                 01 E8(2F)
                                                                                           1E DB 03
                                                                                                    40
                                                                                                        57
                                                                                                           95 C9<74>
                                                                                                           14 AE (4B)
                                                     86 E1 AD 12 86
                                                                                   03AB: 9B FØ
                                                                                              3E C9 1F
                                                                                                        FØ
@@T8:55
        18 6D
              92 96
                     BB
                           30(87)
                                         01F0:85 E0
                                                                     29< EE>
0040:EF
        C9
           27
              30 03 40
                        20
                           91<F@>
                                         01F8:7F
                                                 C9
                                                     20
                                                        10 06 18 69
                                                                     48(14)
                                                                                  0380:BE 02 F0 08 C9
                                                                                                        61
                                                                                                           30 07(F3)
                                                                                  03B8:C9
                                         0200:4C
                                                 84
                                                     93
                                                                                                           20
                                                           60
                                                                 23
                                                                     3B(AB)
                                                                                                    38
0048:AD 02 06 F0 E3 AD
                        63
                           96<84>
                                                        C9
                                                              10
                                                                                           7B
                                                                                              10 03
                                                                                                       E9
                                                                                                              18<B1>
                                         0208:E9 28 8D 13 86 A9 80 2C<6B>
                                                                                  03C0: 6D B6
0050:F0
        DE
           A6
              54
                     88
                        SD
                           00<68>
                                                                                              82 AE 20 86
                                                                                                           BE 10(4A)
                           11(2F)
0058:06
        8D 01
              96 E9 89
                        F@
                                         0218:12 R6
                                                     FB 89 AD 13 BA
                                                                    18(02)
                                                                                   03CB:06 EC
                                                                                              22 86
                                                                                                    30
                                                                                                       ØВ
                                                                                                           FO DAKERS
                                         0218:69
                                                        13 66 AD
0060:AD 00 06
              18 69
                    28 8D 00(FF)
                                                 198
                                                     80
                                                                 13
                                                                     86(C2)
                                                                                  03D0:CE
                                                                                           20 96
                                                                                                 18
                                                                                                    90
                                                                                                       ap
                                                                                                           C9
                                                                                                              1F(29)
        90
0068:06
           02
              EE 01 06
                        CA DØ(12)
                                         0220:A0 00
                                                     91 E0 EE 10 06
                                                                     ADKE93
                                                                                   03D8:D0
                                                                                           03
                                                                                              4C 6E 95 20
                                                                                                           A5 92(FB)
                                                                                                        26
                                         8228:10 BA
                                                                                           17
0070: EF
        AD
              00
                  18
                     65
                        55
                                                     CD 92 94 D9 88 494975
                                                                                  83E8:28
                                                                                              95 EE 20
                                                                                                           4C 61<40>
0079:00 OA
           90 03 FF 01
                        RA AF(49)
                                         0230:00 BD
                                                     18 86 EE 11
                                                                 86 AD(26)
                                                                                  03EB: 94 A9 FF
                                                                                                 AD FC
                                                                                                       02 AD 21<FD>
           EB
                                         0238:11
                                                 196
                                                     CD 93 96
                                                              30
                                                                 83
                                                                     20(37)
                                                                                  03F0:06 BD
                                                                                              20 86 A9
                                                                                                        80
                                                                                                              2E(25)
0080:03
        04
              EB
                 AD 00
                        06 18(98)
        58
           65
              EØ
                 AD 01
                                         0240 - 3F
                                                 93
                                                     A0 01 60 AD
                                                                 98 86<68>
                                                                                  0CFB: 99
                                                                                           4C 83 AD
                                                                                                    23 86
                                                                                                           85 E2<A7>
0090:59 85 E1
              A5
                 58 A9 00 85 AF)
                                         B248:19 69
                                                     28 85 E2 AD 81 86(D1)
                                                                                  8490: AD 24 86 85 E3
                                                                                                       AD
                                                                                                           22 86(89)
                                         0250:69 00
                                                     85 E3 AD 88 86 18(F8)
                                                                                  0408:38 ED
                                                                                              21 06 A8 B1
                                                                                                           E2 00(83)
0098:F2
        49
           98 85 E3 20 60 92<29>
                                         0258:69 50
00A0: A9
        90
           80
              10
                 66
                     SD
                                                     85
                                                        EØ AD 01
                                                                  66
                                                                     69(31)
                                                                                   0410:09
                                                                                           Œ
                                                                                                 26
                                                                                                    88
                                                                                                        10
                           86<96>
                        11
        2E A9 00 99 40
91 20 92 92 A0
80A8: A4
                        83 20<97>
                                         0268:00 05 E1 A5 E2 18 65 58(80)
0268:05 E2 A5 E3 65 59 85 E3(80)
                                                                                  841B: E1
                                                                                           95 4C 9E 95 AØ 8Ø BC(11)
                                                                                  0420:1F
                                                                                           DØ 8C ØA D4 C8 DØ F7<36>
0000:02
                  92 AR RI ARK 983
           92
0089:20
        86
              AD
                 00 06
                        85
                           EØKCB>
                                         0270:A5
                                                 EØ
                                                     10
                                                        65
                                                           58 85
                                                                     A5<712
                                                                                   0428:60
                                                                                           AD
                                                                                              23
                                                                                                    85
                                         0278:E1 65 59 85 E1 AE
0000:AD 01 06
              85 E1 A5 E8 18 DB>
                                                                  83 86(57)
                                                                                  0430:06 85 E3 AD 20 06
                                                                                                           38 EDKFAX
                                                                                  0438:21 06 48 81
                                         0280: CA FB
                                                     83
                                                        28 58 92
                                                                 AE Ø3KEF>
                                                                                                    £2
                                                                                                       49
                                                                                                           88
                                                                                                              91(18)
00CB: 65
        58 85
              EØ
                 90 02
                        E6 E1<21>
        E1
0000:A5
                                         B288: 25 AD
                                                     88
                                                        96
                                                           18 65
                                                                  58
                                                                     85<E5>
                                                                                  0440:E2
                                                                                           60
                                                                                              ΑE
                                                                                                 20
                                                                                                    86
                                                                                                        EC
              65 59 85 E1 AB(E3)
           81
0008:00 A9 51
              91
                 EØ 20
                        22
                           92<8B>
                                         0290:E2 AD 01 06 65 59 85 E3<F0>
                                                                                  0448:D0 03
                                                                                              4C 61 94 CA 8E 20(81>
00E0:C8 A9 45
              91 EB A9 88 BD<17>
                                         8298: A5 E2
                                                     18 69
                                                           28 85 E2 99(18)
                                                                                  2450:26
                                                                                           BE 10
                                                                                                 86
                                                                                                    69
                                                                                                       20
                                                                                                           28 ASKE40
                                         82A0:82 E6
                                                     E3
                                                        CA 00 F2
                                                                 69
                                                                     00(89)
                                                                                  0458:92
                                                                                           20
                                                                                              17
                                                                                                 95
                                                                                                    40
                                                                                                        61
                                                                                                           94
                                                                                                              AE(26)
00EB: 04 06 EE
              04 06 AD 00 A5 BE>
           69
20F2: E2
        18
              28 85 E0
                        98
                           02(50)
                                         02A8:A8 00 C8 91
                                                           E2 CC 82 86<57>
                                                                                  0460:20 06
                                                                                              EC 21 06 00 03 4C(4A)
           49
              70
                 91 EB
                                         0280:00 FB AE 03 06 CA 8E 11(07)
                                                                                  8468:61 94 CA BE
                                                                                                    28
                                                                                                       86
                                                                                                           28
                                                                                                              17(20)
00F8:E6
        E1
                        20
                           20<8A>
0100:92 C8 A9
              7C 91 EØ AD Ø3<C4>
                                         8288:86 68 A6
                                                        2E BD 4C 83 38(7C)
                                                                                  2470:95
                                                                                           40
                                                                                              å1
                                                                                                 94 AE
                                                                                                        20 06 E8<00.
              06 D0 DC
                        A5 E0 40>
                                         8200:10 CD 82
                                                        86
                                                           10 18
                                                                 BC
                                                                     4EKCB3
                                                                                   8478:8E 20
                                                                                              06 20 17
                                                                                                        95
                                                                                                           40
                                                                                                              61<95>
        CD 04
0108:06
                                         02CB: 83 38
                                                     13 CC 03 06
                                                                  18 8E(D7)
                                                                                  2480:94 AD B6 02 49 80 8D 86<2C>
0110:10
        69
           28
              85
                 EØ
                     90
                        82
                                                                                  0488:02 20 17
                                                                                                 95 4C 61
0118:F1 A0 00 A9 5A 91 E0 20<3E>
                                         0200:80 10 06 8C
                                                           11 86 A9 88(8A)
                                                                                                           94 AD(79)
        92
              A9 43 91
                                         02D8:9D
                                                 4C 63 A6
                                                           01 60
                                                                  89
                                                                     00(75)
                                                                                  0490: BE
                                                                                           82 F8
                                                                                                 05 A9
                                                                                                           40
                                                                                                              95<39>
                                                                                                        90
0120:22
           CD
                        EB 68(AB)
                                         82E0:90 40
                                                     82 A8 8D 68
                                                                                   0498:95 A9 40 8D BE
                                                                                                           20
                                                                                                              17(B3)
0128:A9
           C8
              91 EØ CC
                           06(91)
                                                                  AD
                                                                                                       22
                        02
                           EØ (BE)
                                                                                  84A8:95 4C 61 94 AD 28 86 CD<7A>
0130:D0 F8 60 A9 00 C8 91
                                         02E8:06 F0 63 4C 9E 95 A9 01<9C)
                                         82F0:8D 29
                                                     86 84
                                                           ZE AD
                                                                 18 86<8A>
                                                                                   04A8:22 06
                                                                                              90 02 00
                                                                                                        39
                                                                                                           38
                                                                                                              ED(32)
0138:CC 02 06 D0 F8 60
                        20 88<7A>
                                                                                   04R0:21
                                                                                              AB AD
0140:92
        AD
           00 06 18 65
                                         02F8:0D 20
                                                     66 80
                                                           21 66
                                                                  18
                                                                     79(9D)
                                                                                           86
                                                                                                        96
                        58
                           B5<5E>
                        85 E3<58>
                                         03-00:4C 03 38 E9 01 30
                                                                 18 CD< 383
                                                                                   9488:AD 24 86 85 E3 B1
                                                                                                           E2 8D(67)
0148:EZ AD 01
              96
                 65 59
                                         0300:02 06
                                                     18 8B 8D 22 86 89(F9)
                                                                                  04C0:26 06
                                                                                                        40
                                                                                                           80 06<07>
                                                                                              29
                                                                                                 7F
                                                                                                    C9
0150:A9 00 85 E0 A9 98 85 E1<E3>
                                         0310:4C 05 F0
                                                        65 4C
                                                              18
                                                                                   84C8:18 69
                                                                                              20
                                                                                                 4C CF
                                                                                                        95
                                                                                                           C9
0158: AE 03 06 EB EB 20 60
                           92(43)
                                                                                              38 E9 40 AB AD 26<22>
80 BC 26 06 00 26<78>
                                         0318:02 06 CA 8E 22 06 A5 58(9D)
                                                                                   04D0:80 03
0160:20 92 92 A6 01 60
                       AB FF(CB)
                                                                                   0408:06 29
                                         8320:18 AD 00 86 BD 23 86 A54F)
@168:C8 B1 F@ 91 F2 CC M2 M6<18>
                                         0328:59 6D 01 06 0D
                                                              24
                                                                  86
                                                                     ACK 633
                                                                                   04E0:06 EE
                                                                                              20 06 AB
                                                                                                        01
                                                                                                           60
0170:00 F6 C8 81 E0 71
                        E2 A5<95>
0178:EØ 18 69 28 85 EØ
                           @2<E@>
                                         0330:11 86 C8 AD 23 86 18 69(AE)
                                                                                   84E8:08 8D 29 96 A7 98 CE
                                                                                                              MOVEED
                        92
0180:E6 E1 AS E2 18 69
                        28 B5(10)
                                         80338:28 80 23 84 98 83 EF 24(84)
                                                                                  24F9: 24 29 45 92 20 17 95 49(C1)
                                                                                   84FB: 98 A0 81 40 A2 08 A0
                                                                                                              21(89)
                                         2340:06 88 D0 EF AD 23
                                                                     18<6A>
0188:E2 90 02 E6 E3 CA D0 D6<22>
                                                                  86
0190:60 A0 00 A5 50
                    91
                        SE 60< 78>
                                         8349:60 10 86 BD 23 86 AD 24<00>
                                                                                   0500:90 1A 03 00 02
                                                                                                       18 60 E8KE5>
Ø198: AC 00 B1
                        84 54(79)
              58 85 50
                                         RCSR-RA AS RR RD 24 RA FE 23(C4)
                                                                                   05.00-FB FR BR BR BR
                                                                                                       DØ F1
                                                                                                              KRCKA)
                                                                                                 90 10 03 09
                                         0358:06 D0 03 EE 24 06 AD 11<38>
                                                                                  9519:68 A9 57
                                                                                                              BA<A3>
0140-84 55 45 58 85 5F 45 59(84)
                                                                                   0518:9D 1B 03 A9 91
                                         0360:06 8D 27 06 EE 02 06 AD<02>
                                                                                                        90
                                                                                                           1D 83<DC>
              C9
                 9B DØ Ø3 4C<13>
@1A8:85 SF 6@
0180:29 93 C9 7D 00 0E 20 82<C4>
                                         0368:27 06 80 11 06 20 23 95<51>
                                                                                   0520:18 68 A0 01 60 68 80 08<96>
                                                                                  Laenge 1318 Bytes
Listing 1. Die »Windows«-Betriebssystemerweiterung. Bitte mit AMPEL eingeben.
```

```
32035 OPEN #1,4,0,"D:WINDOS.OBJ":FOR F=1
32000 DATA_104,104,141,85,3,104,141,84,3
                                                       TO 6:SET #1,HI:NEXT F
,104,141,89,3,104,141,88,3,169,7,162,16,
                                                                                                  (CC)
32,86,228,96
                                            CNJD
                                                      32040 HI=USR(ADR(BLDAD$),37120,1400)
                                                                                                  <QL>
32010 RESTORE 32000:DIM BLOAD$ (25)
                                            (US)
                                                      32045 CLOSE #1
                                                                                                  <RM>
32020 FOR F=1 TO 25:READ HI:BLOAD$(F,F)=
                                                      32050 HI=USR (37120)
                                                                                                  <TZ>
                                            <UA>
CHR$(HI): NEXT F
                                                      32068 END
                                                                                                  (KS)
32030 POKE 106,144: GRAPHICS 0
                                            <BU>
Listing 2. Basic-Lader für »Windows« (bitte mit dem Atari-Prüfsummer eingeben)
```

```
5 REM BEISPZ.BAS
                                                      555 GOSUB 1100:LX=22:LY=1:POSITION 8,0:G
                                                      OSUB 2300:2 #3; "_Wester_mat_(CTL Y)RETUR
10 REM
                                             <J0>
                                                            Y}.";:INPUT #3;I#
                                                                                                   (BD)
20 REM Zuerst "WINDINIT.BAS" laufen lass
                                                      NICCEL
                                             CICLS
                                                      557 CLOSE #3
                                                                                                   <NE>
                                                      540 IF SEW >0 THEN GOSUB 1700: SOTO 502: RE
KO REM
                                             <.70>
100 REM *** 23 ***
                                             <IW>
                                                      M >>> GEWONNEN <<<
                                                                                                   <JX>
    ? "(ESC CTL ()":SETCOLDR 2,5,4:SETCO
                                                      570 NEXT N
                                                                                                   < IH>
LOR 1,0,14: SETCOLOR 4,5,0
                                            <TY>
                                                      580 GOTO 502
                                                                                                   < MO>
200 DIM N$ (40), I$ (20), NA$ (20), PU(2), AW$ (
                                                      997 REM
                                                                                                   <CD>
1),X(2),Y(2),GF(2)
                                                      998 REM UP FRAGE
                                            (CE)
                                                                                                   <HS>
210 N$="_":N$ (40) "_":N$ (2) -N$ (1)
                                             (SS)
                                                      999 REM
                                                                                                   (CJ)
   X(1)=0:X(2)=18:FU(1)=0:PU(2)=0
                                                      1000 7 #3; "JetztabistaDuadran, ":? #3; N$(
220
                                             (JT)
238 CF(1)=0:CF(2)=0
                                             (VC>
                                                      N#28-19,N#28);
                                                      1010 ? #3; "WillstaDuweineaKarteziehena(J
500 GOSUB 1200: REM >>> ANFANG <<<
                                            <DV>
                                                      /N) _7_";
                                                                                                   <KG>
502 FOR N=1 TO 2
                                             <NU>
510 LX-20:LY=7:POSITION X(N),N=2-1:GOSUB
                                                      1020 RETURN
                                                                                                   <0Z>
                                                      1097 REM
                                                                                                   < VB>
 2300
                                                      1098 REM UP PUNKTEANZEIGE
                                                                                                   (LE>
520 GOSUB 1000
                                             <000>
530 IF CF(N) ≈1 THEN GOSUB 2100:GOTO 550
                                            <EG>
                                                      1899 REM
                                                                                                   (VJ)
                                                      1100 LX=30:LY=10:POSITION 4,10:GOSUB 230
540 GOSLIB 2000
                                             (RE)
550 GOSUB 1600:REM >>> KARTE ZIEHEN <<<
                                            (UR)
553 GOSUB 1800
                                            <UX>
                                                      Listing 3. Ein einfaches Spiel mit Windows
```

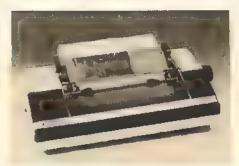
@	<eh></eh>	1730 ? #3:? #3; "Naechstes_Spiel_mit_{CTL	
1110 7 #3:7 #3: "*****23_****"	<si></si>	YERTURNICTL YE."	<u.d></u.d>
1120 7 #3:7 #3;N\$(1,20);":_";PU(1):? #3:		1740 INPUT #3; I\$:PU(1)=0:PU(2)=0:GEW=0:C	
7 #3; N\$ (21,40); ": _"; PU(2)	<xm></xm>	LOSE #3	<2R>
1130 ? #3:? #3:? #3:"(C)_1985_DWS"	<hc></hc>	1750 RETURN	<pw>&gt;</pw>
1140 RETURN	<ph></ph>	1797 REM	<vp></vp>
1197 REM	< UD>	1798 REM PRUEFEN. OB SPIEL ZU ENDE	<jd></jd>
1198 REM UP ANEANG	<tq></tq>	1799 REM	<vx></vx>
1199 REM	<vl></vl>	1900 IF PU(1)(21 DR PU(2)(21 DR GEW)0 TH	
1200 LX=32:LY=10:POSITION 3,0:60SUB 2300		EN RETURN	KEE>
1210 ? #3:? #3: "*****************	<qz></qz>	1810 IF PU(1)>PU(2) THEN GEW-1: RETURN	<zh></zh>
1230 ? #3:? #3; "WiewheisstwderwerstewSpi		1820 IF PU(2)>PU(1) THEN GEW=2:RETURN	KAID
eler_7":INPUT #3:I\$:N\$(1,20)=I\$	<aj></aj>	1830 GEW=3: RETURN	<aw></aw>
1235 IF Is="" THEN NS(1.5)="ATARI":CF(1)	****	1997 REM .	<vt></vt>
=1	<cf></cf>	1998 REM UP SPIELERZUG	<ds></ds>
1240 ? #3:7 #3:"WiewheisstwderwzweitewSp		1999 REM	<wb></wb>
ieler_?":INPUT #3:I\$:N\$(21.40)=I\$	<ga></ga>	2000 X=15:Y=3:POINT #3,X,Y:INPUT #3:AW\$	<gj></gj>
1245 IF I ="" THEN N# (21,28) = "COMPUTER":	1	2020 IF AW\$="N" THEN KG=0: RETURN	<dn></dn>
CF(2)=1	<2K>	2038 IF AW\$<>"J" THEN 2000	<ly></ly>
1250 CLOSE #3:RETURN	(EQ)	2040 KG=1:RETURN	<ae></ae>
1597 REM	<vl></vl>	2097 REM	<vc></vc>
1598 REM UP KARTE ZIEHEN	<lh></lh>	2098 REM UP COMPUTERZUG	<ex></ex>
1599 REM	<tv></tv>	2099 REM	<vk></vk>
1600 IF NOT KG THEN RETURN	<pe></pe>	2100 GDSUB 2200	<xy></xy>
1610 PP=INT(RND(0) +3)+3	<ml></ml>	2110 IF KG=1 THEN ? #3: "J": RETURN	<wi></wi>
1615 PU(N)=PU(N)+PP	<tk></tk>	2120 ? #3: "N": RETURN	<8F>
1620 7 #3: "Du_hast_":PP: "_Punkte":7 #3:"		2197 REM	<ve></ve>
gezogen."	<s6></s6>	2198 REM UUP COMPUTERENTSCHEIDUNG	<dy></dy>
1630 IF PU(N)>23 THEN ? #3: "Du_hast_verl		2199 REM	<mv></mv>
oren!": GEW=N+(N=1) - (N=2)	<mh></mh>	2200 IF PU(N) < 19 THEN KG=1: RETURN	<xu></xu>
1635 IF PU(N)=23 THEN ? #3: "Du_hast_gewo		2210 IF PU(N)>20 THEN KG=0: RETURN	<at></at>
nnen!":GEW=N	<0H>	2220 IF PU(N+(N=1)-(N=2))<18 THEN KG=0:R	
1640 RETURN	<pr>&gt;</pr>	ETURN	<vz></vz>
1697 REM	<vn></vn>	2230 KG=1:RETURN	<af></af>
1698 REM UP ENDE	(BE>	2297 REM	<v6></v6>
1699 REM	<vv></vv>	2298 REM LUP WINDOW DEFFNEN	<fa></fa>
1700 LX=34:LY=6:POSITION 2,2:GOSUB 2300	<um></um>	2299 REM	<0V>>
1705 IF GEW=3 THEN ? #3; "Unentschieden!"		2300 POKE 1538, LX: POKE 1539, LY: POKE 880,	
:? #3:GOTO 1730	<mj></mj>	255: OPEN #3,12,0,"W: ": RETURN	<yb></yb>
1710 ? #3; "Bravo,,"; N\$ (GEW=20-19,GEW#20)	<ak></ak>		
1720 2 #3; "Du_hast_gewonnen."	KLM>	Listing 3. Ein einfaches Spiel mit Windows (Schluß)	)

# Großes Preisausschreiben

Dieses Sonderheft haben wir für Sie gemacht, liebe Leser. Und damit Sie in Ihren Sonderheften nur solche Themen vorfinden, die Sie auch wirklich interessieren, brauchen wir Ihre Meinung. Schließlich soll ein mögliches nächstes Atari-Sonderheft noch besser, interessanter und informativer werden. Und damit wir wissen, was Sie wotlen, schicken Sie bitte den ausgefüllten Fragebogen an uns zurück.

Natürlich gibt es auch etwas zu gewinnen. Unter allen Einsendungen verlosen wir viele interessante Preise.

Um an der Verlosung teilzunehmen, müssen Sie den folgenden Fragebogen ausfüllen, ihn in ein Kuvert stecken und an unsere Adresse schicken:





- 2 Atari 1050-Diskettenlaufwerke, gestiftet von Atari
- 1 Atari 1029-Drucker, gestiftet von Atari
- 10 Gutscheine über je ein Buch freier Wahl aus dem Markt & Technik-Angebot
- 5 Textverarbeitungs-Programme Atari-Schreiber für alle Atari-Computer, gestiftet von Atari
- 20 T-Shirts im »Happy-Computer«-Look

Redaktion Happy-Computer, Markt&Technik Verlag AG, Kennwort. Atari-Sonderheft, Hans-Pinsel-Straße 2, 8013 Haar Der Rechtsweg ist ausgeschlossen, Einsendeschluß ist der 30. April 1986.

(wb)

### Fragebogen zum Atari-Sonderheft von Happy-Computer

Wie hat Ihnen dieses Sonde	erheft Insgesal	mt gefallen?		Möchten Sie sich einen Atari ST kaufen?
☐ Sehrgut ☐ Gut ☐ I	Mittel   Wei	niger gut 💢	Gar nicht	□ ja. □ nein
				Falls nein, warum nicht?
Wie fanden Sie die einzelne	n Rubriken?			and the state of t
(zutreffendes bitte ankreuz				
	Sehr			
	interessant	interessant	uninteressant	
Hardware-Bastelelen				
Software-Tests				
Hardware-Tests				Falls ja, warum?
Spiele-Test				
Anwendungs-Listings				
Spiele-Listings				
Tips und Tricks-Listings				
Turbo-Basic-Interpreter				
Turbo-Basic-Compiler				
Turbo-Basic-Tell (aligemein)			, 🗅	
				Welche Peripherie besitzen Sie für Ihren Ateri-Computer?
Weiche Beiträge haben ihne	en am besten g	etalien?		☐ Diskettenlaufwerk, Typ?
1				□ Kassettenrecorder, Typ?
				□ Druckerinterface, Typ?
2				
3				Drucker, Typ?
·			_	□ Monitor
				O S/W O Farbe Typ?
4				☐ Fernsehgerät  ○ S/W ○ Farbe
5				○ S/W ○ Farbe
				Wetche Peripherie wollen Sie sich 1986 noch kaufen?
3				•
				☐ Diskettenlaufwerk, Typ?
Was haben Sie in diesem So	anderheft vermi	iA+2		☐ Kassettenrecorder, Typ?
HE HESEN SIE IN GEGEGNE SQ	Maritali term	IDE:		☐ Druckerinterface, Typ?
				□ Drucker, Typ?
				☐ Monitor
				O S/W O Farbe Typ?
				□ Fernsehgerät
				OS/W OFarbe
				5 0.44 5 Tabe
				Maria has Branch and a series of the file of the series of
				Weiches Peripheriogerät würden Sie, falls Sie eines gewinnen, bevorzugen?
				☐ Atari 1050-Diskettenlaufwerk
				☐ Atari 1029-Drucker
				Welche Computer-Zeitschriften lesen Sie regelmäßig?
Nie oft würden Sie sich ein :	Sonderheft zur	n Thema Ater		
100XL/130XE kauten?				
🗆 Vierteljährlich 💢 🖂 Halbjäl	hriich 🗆 Einr	nal im Jahr		
Velchen Computer besitzen	ei-a			
	260 ST 🗆	520 ST+		
einen anderen, welchen?				
laben Sie vor, sich 1986 eine	en enderes Co	mandae an ber	dan?	Ich bin damit einverstanden, daß die hier gemachten Angaben elektronisch
	en anderen Co	mputer zu Kat	nen?	verarbeitet werden.
] ja 🔲 nein				Anschrift:
Venn ja, weichen?				
260 ST				
3520 ST+				
∃ Amiga				
MS-DOS-Computer (z.B. IB	M-PC. PC 10)			
				Alter: Jahre
				F WOOD WINNESS



#### Mein Atari-Computer



#### Zielgruppe: Einsteiger und Fortgeschrittene

Das Buch »Mein Atari-Computer« ist als Standardwerk für den Einsteiger anzusehen. Von den elf Kapiteln widmen sich fünf der Hardware, Es wird zunächst der Aufbau der älteren Modelle (Atari 400 und 800) besprochen; im Anhang findet man noch Erklärungen ZIJM. 800XL. **Allerdings** schenkt man den älteren Computern und Peripheriegeräten in diesem Kapitel allzuviel Aufmerksamkeit. Es werden teilweise Geräte beschrieben, die nicht mehr im Handel sind oder die es in Deutschland nie gegeben hat. Die neuen Computer kommen also zu kurz, Immerhin sind die Anschlüsse der Computer beschrieben, damit man weiß, welche Zusatzhardware sich an welchen Aus-/Eingang anschlie-Ben läßt.

Es lassen sich aber eine Reihe von Parallelen ziehen. Beispielsweise unterscheidet sich der Umgang mit dem alten Atari 810-Laufwerk nur in einigen wenigen Details vom neuen 1050-Laufwerk. Auch der Umgang mit dem alten Programmrecorder 410 entspricht dem mit dem 1010-Recorder, Man fühlt sich als Besitzer eines neuen Computer und Peripheriegerätes also nicht unbedingt vernachlässigt.

Interessiert sich jemand für Grafik, dann findet er in den Kapiteln »Einführung in die Grafikfunktionen des Atari-Computers« und »Weiterführende Beschreibung der Grafikfunktionen des

Atari-Computers« ■ wirklich ausführliche Informationen. In den beiden Kapiteln widmet sich der Autor auch der Player Missile-Programmierung, mit der bekanntlich bewegte Grafik erzeugt wird. Damit aber das Beschriebene noch verdeutlicht wird. hat man einige sehr gut dokumentierte Programmbeispiele. miteingebaut. Schließlich läßt sich in der Praxis doch so manches besser nachvollziehen als mit bloßer Theorie.

Weiterhin beinhaltet das Buch auch noch einiges über Musik-Programmierung, Joystickabfrage und Paddles, das Diskettenformat und wie man sich eine eigene Dateiverwaltung aufbauen kann. Auch zu diesen Themen gibt es Programmbeispiele.

Das letzte Kapitel wendet sich schließlich jedem einzelnen Basic-Befehl und jeder einzelnen Basic-Funktion zu. Hier findet man auch Informationen zu den verschiedenen XIO-Funktionen, denen im Handbuch zum Computer kaum Aufmerksamkeit geschenkt wird

Der Anhang informiert den interessierten Leser noch über eine Reihe von PEEKs und POKEs, die der etwas fortgeschrittenere Programmierer unbedingt kennen sollte. Auch die Speicheraufteilung der alten sowie der neuen Computer erläutert ein eigenes Kapitel

Das Buch »Mein Atari-Computer« ist ein rundum gut aufgemachtes und informatives Buch. Bei der-Arbeit stört allerdings sehr, daß Stichwortverzeichnis kein existiert. Sucht man bestimmte Informationen, muß man nicht selten das Buch von vorne bis hinten durchblättern. Es wäre wünschenswert, diesem Buch zumindest einige Seiten mit einem separaten Stichwortverzeichnis beizulegen. Vielleicht geht dieser Verbesserungswunsch ja schon bei der nächsten Neuauflage in Erfüllung.

on Poole, Martin McNith, Steven Cook → Mein Atari-Computers, te-wi Verlag, ca. 470 Setten, ISBN 3-921803-18-7, Preis 59 Mark

#### Das Atari-Programmierhandbuch



Zielgruppe: Einsteiger

Wenn Sie noch nie programmiert haben, aber einen Atari-Computer besitzen oder kaufen möchten, dann bietet sich »Das Atari-Programmierhandbuch« für den optimalen Elnstieg an. So beschreiben die ersten Kapitel, was ein Programm überhaupt ist und wozu es dient. Langsam wird man dann an sein erstes Programm herangeführt.

Auch macht das Buch den Atari-Besitzer von vornherein mit der richtigen Programmiertechnik vertraut. Anhand von Flußdiagramzeiat die Autorin anschaulich den systematischen, überlegten Aufbau eines Programms. Solche Diagramme dienen schließlich auch dazu, Programme möglichst optimal zu gestalten. So spart man sich einige Wiederholungen und das Programm wird gleichzeitig uberschaubarer.

Neben ausführlichen Erklärungen des Atarı-Basic-Befehlssatzes und Erläuterungen zu den Fehlermeldungen, wird auch die Unterprogrammtechnik sowie die Stringverarbeitung angesprochen.

In »Das Atari-Programmierhandbuch« kommen Programmbeispiele nicht zu kurz. Insgesamt sind es 55. Man muß sie aber nicht unbedingt abtippen, da man sämtliche Programme auch auf Diskette beziehen kann.

(wb)

Linda M. Schreiber »Das Atari Programmierhandbuch«, Markt & Technik 403 Seiten SBN 3-89090-062-3. Prais 62 Mark

#### Das Atari-Buch, Band 2



#### Zielgruppe: Fortgeschrittene und Profis

Es ist naheliegend, daß man sich nach dem Kauf seines Atari-Computers zunächst mit der Basic-Programmierung beschäftigt. Schließlich liegt das Basic gleich nach dem Einschalten des Computers vor. Hat man seinen Computer unter Basic aber ausgereizt, steht Maschinensprache-Programmierung an. Schließlich ist nichts schneller als reine Maschinensprache. Hinzu kommt, daß sich gewisse Funktionen des Atari-Computers nur von Maschinensprache aus realisieren lassen.

Für »Das Atari-Buch, Band 2« werden zumindest Basic-

Programmier-Kenntnisse vorausgesetzt. Sollten Sie sich mit Maschlnensprache bereits ein wenig beschäftigt haben, ist dies auch vorteilhaft. Für die Einführung in die Maschlnensprache-Programmierung sind am Ende des Buches ein Editor, ein Assembler und ein zugehöriger Disassembler abgedruckt. Somit kann man auch alle im Buch aufgelisteten Maschinenprogramme gleich ausprobieren.

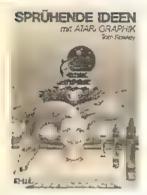
In einem eigenständigen Kapitel werden die Interruptfähigkeiten der Atari-Computer erschöpfend behandelt. Mit dieser Technik lassen sich einige Programmvorgänge unabhängig vom restlichen Programm ausführen. Besonders wichtig ist die Interrupttechnik in Spielen.

Das letzte Kapitel widmet sich der Display-List, mit der man die Bildschirmausgabe beeinflussen kann. Weiterhin wird in diesem Kapitel auch waagrechtes und senkrechtes Scrollen besprochen.

Wer also mehr aus seinen Programmen machen möchte, sollte auf »Das Atari-Buch, Band 2« zurückgreifen. Schließlich steckt im Atari mehr als nur ein eingebautes Basic. (wb)

Hans Lorenz Schneider, Rudolf Bichler, »Das Atari-Buch, Band 2«, Markt & Technik, 197 Seiten. ISBN 3-89090-072-0. Preis 32 Mark

## Sprühende Ideen mit Atari-Graphik



#### Zielgruppe: Anfänger und Grafikinteressierte

Mit einer Auflösung von 320x192 Punkten und bis zu 256 gleichzeitig darstellbaren Farbschattierungen, ist der Atari-Computer kaum zu schlagen. Allerdings ist Grafikprogrammierung die nicht auf Anhieb verständlich. Vor allem wenn man seine Programme mit bewegter Grafik ausschmükken möchte, wird es kompliziert. Die Player Missile-Grafik, die für Bewegung auf dem Bildschirm sorgt, läßt sich nämlich nur mit entsprechenden Kenntnissen programmieren.

Wer animierte Grafik einsetzen, mehr über die verschiedenen Grafikmodi des Atari-Computers wissen, eine neue Display-List erzeugen und einen neuen Zeichensatz definieren möchte, findet zu diesen Themen jeweils ein eigenes Kapitel vor. Um Ihr neu angeeignetes Wissen auch gleich zu testen, wurden in das Buch Fragen eingebaut, die am Ende des Buches beantwortet werden

Im Anhang finden sich noch einige Arbeitsblätter zum Entwurf und zur Gestaltung von Bildschirmen. Diese Seiten sollte man sich natürlich fotokopieren, um diese wirklich sinnvollen Hilfen immer wieder einzusetzen.

Tom Rowley: "Sprühende Ideen mit Atzri Graphik«, te-wi Verlag, 211 Seiten, ISBN 3-921803-39-X, Pres 49 Mark

## Atari 600XL/800XL



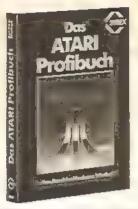
### Zielgruppe: Fortgeschrittene und Profis

Wenn Sie sich bereits in Basic auskennen, sich anschließend der Maschinensprache zuwenden und gleichzeitig natürlich auch mehr über die Hardware des Atari-Computers wissen möchten, dann ist das Buch Atari 600XL/800XL Intern« für Sie ein unerlaßliches Hilfsmittel. Angefangen von der Speicherverteilung, bis hin zu ausführlichen Erklarungen der einzelnen Spezialbausteine des Atari-Computers, findet man alles Wissenswerte. Ausführlich wird jedes einzelne Register, jeder wichtige Baustein besprochen, sei es der Antic (zuständig für Grafik), der GTIA (Bildausgabe) oder der Pokey (zuständig für Tonausgabe, Tastaturabfrage etc.).

Im letzten Kapitel ist noch der Gesamtspeicherplan der Atari-Computer abgedruckt, der nochmals alle Adressen genau beschreibt. Wer also wirklich alles über seinen Atari wissen möchte, für den ist das Buch »Atari 600XL/800XL Intern« ein absolutes Muß. (wb)

Eichler Grehmann »Atan 600XL/800XL Intern« Data Becker 383 Seiten ISBN 3-89011-053-3. Preis 49 Mark

## Das Atari Profibuch



#### Zielgruppe: Fortgeschrittene und die es werden möchten

Abgesehen von der »etwas anderen Einleitung« mit
viel Humor, verbergen sich
im »Atari Profibuch« Informationen, auf die man zukünftig
nicht mehr verzichten kann.
In diesem Buch findet man
alles über Speicheradressen, die speziellen Bausteine und die Besonderheiten des Atari.

Vor allem den Besonderheiten der Atari-Computer haben sich die Autoren in diesem Buch zugewandt. So ist ein Hauptaugenmerk auf die Funktion des Antic-Bausteins und die verschiedenen GTIA-Grafikmodi gerichtet. Auch Informationen darüber, wie man Bildschirmscrollen realisiert, wie man den Display List Interrupt verändern kann und wie Interrupts allgemein anzusprechen sind, fehlen nicht.

Wer seinem Atari-Computer schon immer mal besser klingende Töne entlocken wollte, findet zu diesem Thema ein eigenständiges Assembler-Programm VOI. Und wen zuletzt noch die Anschlüsse der Atan-Computer interessieren. kann sich am Ende des Buches über eine Reihe von Bildern freuen, die jeden einzelnen Anschluß genauestens erklären. Sogar einen Schaltplan des 800XL kann das »Atari Profibuch« aufweisen. Mit diesem Buch werden einem also alle Tore, nicht nur zur Hard varewelt, weit geöffnet.

## 6502-Assembler-Kurs für Beginner



#### Zielgruppe: Fortgeschrittene

Schon das Äußere des Buches weicht von der Norm ab. Es ist kein starres, gebundenes Buch, sondern eher als Paperback zu bezeichnen. Es bezieht seine Angaben auf den Assembler T.EX.AS (= Terminal Extended Assembler), welcher ebenfalls aus dem Hause Dripke stammt.

Das Buch ist sowohl für die Theorie als auch für die Praxis geschrieben, so daß man es am besten am Computer bearbeitet. Die Assemblerbefehle werden in kleinen Programmen vorgestellt, so daß die Wirkung aller Befehle sofort ausprobiert werden kann.

Zwischen diesen mehr praktisch orientierten Kapiteln sind andere mehr theoretischer Art eingestreut, in welchen von Bits und Bytes. vom Speicheraufbau oder von den verschiedenen Adressierungsarten in der Assemblersprache gesprochen wird. Sie sind, wie das ganze Buch, in einer lockeren, leicht verständlichen Sprache geschrieben. Und hat man wirklich einmal etwas nicht verstanden, so kommt am Ende eines jeden Kapitels noch einmal eine Zusammenfassung, in welcher das Wichtigste der letzten Seiten wiederholt wird.

(Arnd Wängler)

Das Atan: Andreas Dinpke, +6502-Assemblerkurs für Beginners, Intertage Age Verlag, ISBN 3-88986-000-1, Pres 29.60 Mark

Reschke, Andreas Wiethoff. »Das Atan Profibuch», Sybex Verlag, 269 Seiter. ISBN 3-88745-605-X, Press 42 Mark



## **Basic-Schalter**

Möchte man das eingebaute Basic abschalten, muß man normalerweise den Atari-Computer erneut booten. Die Maschinencode-Routine »BSWITCH.COM« kann das genauso gut.

Is man sich bei Atari dazu entschieden hat, das eingebaute Basic durch Drücken der OPTION-Taste nur während des Bootens zu desaktivieren, hat man einige Anwender nicht gerade glücklich gemacht. Doch die Notwendigkeit ständig neu zu booten, besteht nun nicht mehr, denn mit der kurzen Maschinensprachroutine BSWITCH.COM« kann man das Basic von DOS aus einbeziehungsweise ausschalten.

Tippen Sie zunächst das kleine Basic-Programm ab, das die benötigte Datei mit dem Namen »BSWITCH.COM« auf der Diskette erzeugt. Sie können nun durch einfaches Laden des Programms »BSWITCH.COM« das eingebaute Basic einoder ausschalten. Geben Sie dazu unter DOS 2.5 zunächst Lein. Betätigen Sie anschließend die RETURN-Taste und laden Sie dann das File »BSWITCH.COM«. War das Basic zu diesem Zeitpunkt eingeschaltet, wird es ausgeblendet; war es ausgeschaltet, funktioniert das Ganze umgekehrt.

Wichtiger Hinweis: Basic funktioniert nur dann einwandfrei, wenn beim Booten nicht OPTION gedrückt wurde!

(Julian F. Reschke/wb)

1000	REM Basic Switch V 1.6	< WW >	
1010	REM von Julian F. Reschke	<28>	
1829	REM (c) Happy Computer	<ev></ev>	
1030	REM	<th></th>	
1040	OPEN #2,8,0,"D:BSWITCH.COM"	<in></in>	
1050	PRUEF=0	<1S>	
1060	FOR I=1 TO 23 .	<gc></gc>	
1070	READ WERT: PRUEF=PRUEF+WERT	<jn></jn>	
1980	PUT #2, WERT: NEXT 1	<fz></fz>	
1090	CLOSE #2: IF PRUEF<>2381 THEN ? "ERR	h	
OR"		<px></px>	
1100	END	(PR)	
1110	DATA_255,255,0,6,10,6,173,248,3,73.		
	1,248,3,76,116,228,224,2,225,2,8,6	<gp></gp>	

### Basic-Listing »BSWITCH«

```
1222
1010 : Basic Switch V 1.0
1020 ; von Julian F. Reschke
1030 ; (c) Happy Computer 1985
1040 :
1050
1060 RUNAD * $02E0
                      Einsprungvektor
1070 BASICF = $03F8
                      ; Basic-Flag
1080 WARMSV = $E474
                      :Warmstartvektor
1070 :
1100
         *= $0600
1110 ;
1120
         LDA BASICF
11 50
         EOR #1
                      :Status 'umdrehen'
1140
         STA BASICF
1150
         JMP WARMSV
                      :Warmstart machen
1160 ;
1170
         *= RUNAD
1180 :
1190
         .WORD $0600 ;Einsprungvektor
1200 ;
1210
         - END
```

Quell-Listing »BSWITCH« (MAC/65-Assembler)

PROGRAMM-STECKBRIEF		
Programmame	Basic-Schalter	
Programmtyp	Utility	
Programmiersprache	Atari-Basic (Assembler)	
Programmlänge	342 Byte	
für Computer	800 XL/130 XE	
zusätzliche Hardware	Diskettenlaufwerk	
Eingabehilfe	Prüfsummer	
Bemerkung	Schaltet den Atari-Basic-Interpreter nach Belieben ein oder aus	
Leserservice	Diskette (BSWITCH BAS)	

## Blitzschnelle Zeichenumwandlung

Die interne Zeichendarstellung unterscheidet sich vom normalen Zeichensatz. Eine Maschinenroutine liest den Bildschirm und ordnet die gelesenen Zeichen blitzschnell einem String zu.

In gemeinsames lästiges Übel der meisten gängigen Heimcomputer ist, daß, je nach Situation, eine andersartige Kodierung von Zeichen benutzt wird. Bei Atari-Computern muß man zwischen zwei verschiedenen Darstellungen unterscheiden.

Die ASCII-Darstellung wird für alle standardmäßigen Einund Ausgaberoutinen des Betriebssystems verwandt. Für den Basic-Programmierer bedeutet dies: Alle Basic-Kommandos wie PRINT, INPUT, LOCATE etc. arbeiten mit dieser Zeichendarstellung.

Daneben gibt es noch die interne Darstellung von Zeichen. Diese findet man beispielsweise bei der Reihenfolge der Zeichen in einem Zeichensatz oder bei der Kodierung der Zeichen im Bildschirmspeicher vor.

Gerade hier setzt das Programm »SCRSTR« an. Häufig gilt es, auf dem Bildschirm dargestellte Zeichen in eine normale Zeichenkette zu übertragen. Das Betriebssystem stellt zwar hierfür eine Funktion zur Verfügung, nämlich den LOCATE-Befehl, der aber auch Nachteile hat: Die einzelnen Zeichen müssen über eine Schleife von der Bildschirmposition in die Zeichenkette übertragen werden. Dies erfordert entsprechend viel Zeit. Solche Verzögerungen sollte man aber bei Programmen, die auf Tastatur-Eingaben reagieren, vermeiden.

Die Maschinenspracheroutine »SCRSTR« überträgt genau 960 Zeichen (40 Zeichen pro Zeite multipliziert mit 24 Zeilen ergibt 960 darstellbare Zeichen auf dem Bildschirm in Grafikstufe 0) eines normalen Textbildschirms in eine Zeichenkette. Drese muß vorher unbedingt auf die 960 Zeichen Länge dimensioniert werden. Teile des Bildschirms können anschließend problemlos als Teile dieser Zeichenkette behan-

delt werden. Hier ein Beispiel (SCREEN\$ ist die benutzte Zeichenkette): Die zweite Bildschirmzeile erhält man mit »SCREEN\$(41,80)«. Wird das 10. bis 20. Zeichen aus der 4. Zeile benötigt, so müssen die Zeichen in SCREEN\$ als Position 130 bis einschließlich 140 gelesen werden.

Noch eine Anmerkung zum Basic-Programm. Nach dem Eintippen speichem Sie das Programm zunächst ab und geben RUN ein. Falls die Daten für die Maschinenspracheroutine fehlerhaft sind, erfolgt eine entsprechende Meldung. Ansonsten wird als Demonstration das Programm geLISTet und in die Zeichenkette SCREEN\$ übertragen, die daraufhin auf dem Bildschirm ausgegeben wird.

(Julian F. Reschke/wb)

PROGRAMM-STECKBRIEF		
Programmname	Screen to String	
Programmtyp	Utility	
Programmiersprache	Basic (Assembler)	
Programmlange	869 Byte	
für Computer	800 XL/130 XE	
zusätzliche Hardware	Diskettenlaufwerk	
Eingabehilfe	Prüfsummer	
Bemerkung	Umwandlung von interner Darstellung in normalen Zeichensatz	
Leserservice	Diskette (SCRSTR.BAS)	

1000 REM	<mx></mx>	
1010 REM Screen to String V 1.0	<zu></zu>	
1020 REM von Julian F. Reschke	<ze></ze>	
1030 REM (c) Happy Computer 1985	<fv></fv>	
1040 REM	<xj></xj>	
1050 REM	<tn></tn>	
1060 REM String fuer Bildinhalt	<kb></kb>	
1070 DIM SCREEN\$ (960):SCREEN\$ (960)="_"	<ve></ve>	
1080 REM Bild fuellen	<vw></vw>	
1090 POKE 82,0:LIST	<wt></wt>	
1100 REM Routine definieren	<wy></wy>	
1110 DIM SCRSTR\$ (91):PRUEF=0	<le></le>	
1120 FOR I=1 TO 91	<ix></ix>	
1130 READ WERT: SCRSTR#(I,I)=CHR#(WERT)	<lb></lb>	
1140 PRUEF=PRUEF+WERT: NEXT I	<pa></pa>	
1150 IF PRUEF<>12436 THEN ? "Fehler!":EN		
D	<yv></yv>	
1160 DATA_104,184,133,215,104,133,214,16		
5,88,133,212,165,89,133,213,24,165,88,10		
5,192,133,216,165,89,105	<hc></hc>	
1170 DATA_3,133,217,160,0,162,0,177,212,		
16,2,162,255,41,127,201,64,16,5,24,105,3		
2,208,9,201,96,16,5,56,233	(QP)	
1180 DATA_64,16,0,224,255,208,2,73,128,1		
45,214,230,212,208,2,230,213,230,214,208		
,2,230,215,165,212,197	<fa></fa>	
1190 DATA_216,208,200,165,213,197,217,20		
8,194,96	<rm></rm>	
1200 REM	<tc></tc>	
1210 REM Routine aufrufen	<gf></gf>	
1228 WERT=USR (ADR (SCRSTR*), ADR (SCREEN*))	<@C>	
1230 REM SCREEN# ausgeben	<kt></kt>	
1248 PRINT SCREEN\$	<th></th>	

Basic-Programm zu »SCRSTR«

```
1000 ;
                                                       LDA (FRØ),Y
                                              1340
1010 ;Screen to String V 1.0
                                              1350
                                                       BPL NICHT_INVERS
1020 ;von Julian F. Reschke
                                              1360
                                                       LDX ##FF
1030 ; (c) Happy Computer 1985
                                              1370 NICHT INVERS
1949 =-
                                              1380
                                                       AND #$7F
1050
                                              1390
                                                       CMP #64
1060 SAVMSC = $58
                       : Vektor auf Anfangs
                                              1400
                                                       BPL GR_64
adresse des Bildspeichers
                                              1410
                                                       CLC
1070 FR0 =
                                              1428
                                                       ADC #32
1080 DEST =
              #DA
                       ; Zeiger auf String
                                                       BNE FERTIG
                                              1430
1090 ENDE =
                       ¿Zeiger auf Ende de
                                              1440 GR 64 CMP #96
s Bildes
                                              1450
                                                       BPL FERTIG
1100 ;
                                              1460
                                                       SEC
1110
             $0600
                       ;Programm ist versc
                                              1470
                                                       SBC
                                                           #64
                                                       BPL FERTIG
hi ebbar
                                              1480
1120 ;
                                              1490 FERTIG
                                                       CPX #$FF
1130
         PLA
                       ;Anzahl der Paramet
                                              1500
                                                                     ; Invers gewesen?
                                              1510
                                                       BNE WAR NICHT
er
1140
         PLA
                       ;Anfangsadresse des
                                              1528
                                                       EOR #$80
                                                                     ; Invertieren
                                              1530 WAR_NICHT
 Strings
1150
         STA DEST+1
                                              1548
                                                       STA (DEST),Y
                                              1550
                                                        INC FRØ
1160
         PLA
1170
         STA DEST
                                              1560
                                                       BNE KEIN CARRY
1180 ;Adr. des Bildsp. holen
                                              1570
                                                       INC FRØ+1
                                              1580 KEIN_CARRY
         LDA SAVMSC
1190
                                                        INC DEST
1200
         STA FRO
                                              1596
         LDA SAVMSC+1
1210
                                              1600
                                                       BNE KEIN_CARRY2
                                                       INC DEST+1
1220
         STA FRØ+1
                                              1610
                                              1620 KEIN_CARRY2
1230 ; Ende des Bildsp. berechnen
                                              1630
                                                       LDA FRE
                                                                     :Ende?
1240
         CLC
1250
         LDA SAVMSC
                                              1640
                                                       CMP ENDE
         ADC # <968
                                              1650
                                                       BNE CONVERT
1260
         STA ENDE
                                                       LDA FRØ+1
1270
                                              1660
1280
         LDA SAVMSC+1
                                              1670
                                                       CMP ENDE+1
1290
         ADC # >960
                                              1680
                                                       BNE CONVERT
         STA ENDE+1
1.300
                                              1690 ;das war's
1310 ; Hauptschleife
                                              1700
                                                       RTS.
1320 CONVERT LDY #0
                                              1710 ;
                                                                     Quell-Listing zu »SCRSTR«
                                                                     (MAC/65-Assembler)
         LDX #Ø
                                              1720
                                                        - END
1330
```

# Bildschirmausschnitt schnell gelöscht

Erfolgen häufig wechselnde Bildschirmausgaben, sollten bestimmte Ausschnitte des Monitors auch gleich wieder frei sein.

anchmal ist es ganz nützlich, wenn man bestimmte Bildschirmbereiche schnell löschen kann. Zwar ist dies durchaus von Basic aus realisierbar, allerdings etwas langsam. Schneller geht es in Maschinensprache. Die kurze Routine »Bildloesch« läßt ein beliebig großes Feld auf dem Bildschirm im Bruchteil einer Sekunde verschwinden.

Die Maschinen-Unterroutine wird durch die Programmzeilen 1000 bis 1120 in den Speicherbereich ab 1536 gePOKEt. Hierbei handelt es sich um die Page 6, die besonders
für kurze Maschinenroutinen geeignet ist. Der Aufruf des
Programms erfolgt mit »X=USR(1536,SC-1+DS,ZEILEN,SPALTEN)«.

Dabei enthält SC die Bildschirmstart-Adresse. Von Basic aus läßt sich dieser Wert auch mit den Adressen 88 und 89 ermitteln. SD stellt die absolute Bildschirmposition dar. Sie wird mit Zeile x 40+Spalte errechnet. ZEILEN gibt die Anzahl der Zeilen und SPALTEN die Anzahl der Spalten an, die das zu löschende Feld umfassen soll.

PROGRAMM-STECKBRIEF		
Programmname	Bildloesch	
Programmtyp	Utility	
Programmiersprache	Maschinensprache	
Programmlänge	690 Byte	
für Computer	800 XL/130 XE	
zusätzliche Hardware	keine	
Eingabehilfe	Prüfsummer	
Bemerkung	löscht Teile des Bildschirms	
Leserservice	Diskette (BLOESCH BAS)	

20 REM DER AUFRUF ERI 30 REM RESSECISIO, SC 40 REM HW WOM HOLFGAN 90 GRAPHICS GILTST 100 GOSUS 1000	OGT HET ZEILEN, SPAKTING IS GZENNY
120 ZETLE=10:SPALTE=: 125 DS=ZETLE#40+SPALT	E >5,10,20) Em Mode EQD D:POKE a,0;
1930 FOP 1190 1100 4,104 1110 251,24,100,176,105,41	184,133,176,16 45,176,136,288,
208, 231, 36	

Hier ist bereits ein Teil des Bildschirms gelöscht

In der vorliegenden Form funktioniert die Routine »Bildloesch« nur in Grafikstufe O. Weiterhin noch in den Grafikstufen, die 40 Byte pro Zeile in Anspruch nehmen. Ansonsten muß der vorletzte Wert in Zeile 1110 entsprechend angepaßt werden. Soll das Programm beispielsweise in Grafikstufe 1 verwendet werden, muß hier der Wert 20 anstatt 40 stehen. (Wolfgang Czerny/wb)

```
(GS)
10 REM LOESCHEN VON BILDTEILEN
28 REM DER AUFRUF ERFOGT MIT
                                              < CM>
   REM X=USR (1536, SC, ZEILEN, SPALTEN)
                                              <KX>
40 REM ** VON WOLFGANG CZERNY
                                              < M(1)
90 GRAPHICS 0: LIST
                                              <HS>
100 GOSUB 1000
                                              <UI)
110 SC=PEEK (88) +256*PEEK (89)
                                              (FS)
    ZEILE=10: SPALTE=5
120
                                              <S0>
125 DS=ZEILE+40+SPALTE
                                              <EC>
    X=USR(1536,SC-1+DS,10,20)
                                              <LY>
130
                                              (NG>
140 GOTO 140
1000 REM UNTERPROGRAMM NACH 1536
                                              <WW>
1010 S=0:RESTORE 1100
                                              <0X>
1020 FOR A=1536 TO 1575: READ D: POKE A.D:
SES+D: NEXT A
                                              <LO>
1030 IF S<>5568 THEN ? "DATEN-FEHLER!":S
                                              < MZ >
TOP
1090 RETURN
                                              <PU>
1100 DATA,104,104,133,177,104,133,176,10
4,104,170,104,104,133,178
1110 DATAL164,178,169,0,145,176,136,288,
                                              (LZ)
251,24,165,176,105,40,133
                                              (KB)
1120 DATA_176,165,177,105,0,133,177,202,
208,231,96
                                              (AS>
Basic-Listing zu »Bildloesch«
```

```
0100 ; ***** Loeschen einzelner
0110 ;**** Bildausschnitte Gr. 0
0120 ;**** X=USR(1536,SCPOS,Y,X)
0130
          Alert .
              $0600
0140
         PLA
0150
         PLA
                       SCPOS NACH
          STA $B1
                       #00B0 UND
0160
2170
          PLA
                       : $00B1
0180
          STA $B0
                       ; ZEILEN IN
0190
          PLA
          PLA
                       : X=REG.
0200
0210
          TAX
                       : SPALTEN
P220
          PI A
9238
          PLA
                       INACH $B2
0240
          STA $B2
0250 L1
          LDY $B2
                       ; $82-YREG.
                       ; LEERZEICHEN
         LDA #2
                       ; SPEICHERN
8280 L2
          STA
              ($BØ)_Y
8299
          DEY
                       ; ZEILE
                       ZU ENDE?
8388
          BNE L2
0.310
          CLC
                       ADDIERE
          1.DA #RØ
0328
                       ; 40 SPLATEN
          ADC #$28
0338
                       ZU SCPOS
8348
          STA #BØ
0338
          LDA #B1
9368
          ADC ##RØ
9379
          STA $B1
                       ; ALLE ZEILEN
9389
          DEX
                       :FERTIS?
Quell-Listing zu »Bildloesch« (MAC/65)
```

# Daten schnell zur Hand

Bilder, Daten und Variablen lassen sich mit »Turbo IO« blitzschnell von Diskette laden.

s ist allgemein bekannt, daß im Betriebssystem des Atari-Computers sämtliche wichtigen Routinen für Ein- und Ausgaben bereits enthalten sind. Unter Basic ist die Auswahl allerdings beschränkt: Man kann die Befehle GET und PUT benutzen, die allerdings immer nur ein einzelnes Zeichen lesen beziehungsweise ausgeben. Eine Alternative stellen die Befehle PRINT und INPUT dar. Allerdings kann man mit diesen Befehlen nur maximal 125 Zeichen auf einen Massenspeicher schreiben und lesen.

Die Funktionen der CIO heißen nun »GET CHARACTERS« und »PUT CHARACTERS«. Sie sind zwar als die Befehle BGET und BPUT in Basic XL., Basic XE und im Turbo-Basic XL definiert, im Atari-Basic fehlen sie aber. Daher soll hier gezeigt werden, wie man sie über eine Maschinensprachunterroutine unter Atari-Basic ansprechen oder simulieren kann.

Für Ein- und Ausgaben stehen auf dem Atari acht »Kanäle« (O bis 7) zur Verfügung. Zu jedem dieser Kanäle gibt es einen sogenannten IOCB (Input/Output-Control-Block). Für uns sind hier allerdings nur drei Register interessant.

ICCOM: Über dieses Register gibt man die Nummer des auszuführenden Befehls an. (Beispiel: »7« bedeutet »PUT CHARACTERS«, »11« bedeutet »GET CHARACTERS«.). Als nächstes interessiert uns die Zwei-Byte-Variable ICBADR, über die die Anfangsadresse des zu lesenden beziehungsweise zu schreibenden Speicherbereiches festgelegt wird. Zuletzt folgt noch ICBLEN. Hier gibt man die Anzahl der zu übertragenden Zeichen an.

Bevor man die Schnelladeroutine anspricht, muß ein Kanal bereits geöffnet sein. Jetzt folgt also nur noch der Sprung nach CIOV (\$E456), und die gewünschte Funktion wird ausgeführt. Der Statuscode der Operation kann übrigens über das Y-Register abgefragt werden. Wir übertragen den Wert in die Variable FRO (\$D4) in Seite O. Über die USR-Routinen können dann die Ergebnisse an das Basic-Programm zurückgegeben werden.

Falls nach RUN die Meldung »Fehler« ausgegeben wird, haben Sie einen Wert in den DATA-Zeilen falsch eingegeben. Korrigieren Sie in diesem Fall bitte das Programm.

Natürlich lassen sich auch Inhalte von Zeichenketten oder Zeichensätzen laden und speichern. Hier sind der Phantasie keine Grenzen gesetzt.

(Julian F. Reschke/wb)

PROGRAMM-STECKBRIEF		
Programmname	Turbo IO	
Programmtyp	Utility	
Programmiersprache	Atari-Basic (Assembler)	
Programmlänge	1146 Byte	
für Computer	800 XL/130 XE	
zusätzliche Hardware	Diskettenlaufwerk	
Eingabehilfe	Prüfsummer	
Bernerkung	Schnelles Laden und Speichern von Daten	
Leserservice	Diskette (TURBOIO BAS)	

	_
1000 REM	<wx></wx>
1010 REM Turbo-1/0 V1.0	<fb></fb>
1020 REM von Julian F. Reschke	<7E>
1030 REM (c) Happy Computer 1985	<fv></fv>
1040 REM	<yj></yj>
1050 DIM TURBO\$(39)	<je></je>
1060 PRUEF=0	<1V>
1070 FOR I=1 TO 39	<la></la>
1080 READ WERT: TURBO\$(I, I)=CHR\$(WERT)	<fx></fx>
1090 FRUEF=PRUEF+WERT	<vh></vh>
1100 NEXT I	<er></er>
1110 IF PRUEF<>3595 THEN ? "Fehler!":END	
	<md></md>
1120 DATA-104,104,104,10,10,10,10,10,170,10	
4,104,157,66,3,104,157,69,3,104,157,68	<xv></xv>
1130 DATA_3,104,157,73,3,104,157,72,3,32	
,86,228,132,212,169,0,133,213,96	<kd></kd>
1140 REM	<mt></mt>
1150 REM Zur Demonstration:	<gy></gy>
1160 GRAPHICS 24: SETCOLOR 2,0,0: COLOR 1	<ah></ah>
1170 FOR I=0 TO 319 STEP 5	< (MD >
1180 PLST I.0:DRAWTO 319-I,191	<lc></lc>
1190 NEXT I	<fs></fs>
1200 FOR I=0 TO 191 STEP 3	<hu></hu>
1210 PLOT 0, I: DRAWTO 319,191-I	<pd></pd>
1220 NEXT I	<ez></ez>
1230 REM	<tl></tl>
1240 REM Bild abspeichern	<bi></bi>
1250 REM Anfangsadresse: PEEK(B8)+256*PE	
EK(89)	<wp></wp>
1260 REM Laenge: 192*40=7680	<lm></lm>
1270 REM Befehl: Schreiben=11	<jc></jc>
1280 GPEN #1,8,0,"D:TEST.PIC"	< PM>
1290 Q-USR(ADR(TURBO\$),1,11,PEEK(88)+256	
*PEEK (89) ,7680)	<mg></mg>
1300 CLOSE #1	<mp></mp>
1310 REM Bild loeschen	<10>>
1320 GRAPHICS 24:SETCOLOR 2,0,0	<tp></tp>
1330 REM Bild laden: Befehlstode 7	<np></np>
1340 OPEN #1,4,0,"D:TEST.PIC"	<ny></ny>
1350 Q-USR (ADR (TURBO\$),1,7,PEEK (88)+256*	
PEEK (89) .7680) : CLOSE #1	<xu></xu>
1360 GOTO 1360	<rw></rw>

Listing 1. »Turbo IO« als Basic-Lader

```
;Turbo-1/0 V1.0
1810
1022
     ;von Julian F. Reschke
SES I
     (c) Happy Computer
1040
1860 ;USR=(ADR (TURBOS), KANAL, KOMMANDO, AD
RESSE, LAENGE)
1079 :
1090 FRB =
1898
     ICCOM = $0342
     ICBADR = $0344
1100
1112
     ICM EN = $0348
1120 CIOV =
             ££456
1130
             4956048
1140
1150
1168
          PLA
                       :Anzahl der Paramet
er
1170
          PLA
          PLA
                       ;Kanalnummer
1180
1190
          ASL A
                        *16
          ASL A
1200
1218
          ASL A
1220
          ASL A
          TAX
1246
          PLA.
1250
                       ;Kommandonummer
          STA ICCOM, X
1260
1270
          PLA
                       : Adresse
1280
          STA ICBADR+1,X
1298
          PLA
1306
          STA ICHADR, X
1318
                       ; Launge
          STA ICBLEN+1,X
1320
1330
1348
          STA ICBLEN, X
1350
          JSR CIOV
                       jab in's OS
13AB
          STV FRA
                       (Fehlercode
1378
          LDA #0
1390
1390
          STA FRØ+1
1400
          . END
1410
```

Listing 2. Quelitext zu »Turbo IO« (MAC/65-Assembler)

# Wortumbruch perfekt

Ohne Rücksicht auf Verluste – sprich Trennregeln – werden unter Basic Worte einfach »zerrissen«. Mit der kurzen Routine »Wordwrap« passiert das nicht mehr.

er schon einmal versucht hat, ein Textadventure in Basic zu schreiben, kennt das Problem: Man kann zwar das Spiel auf dem Bildschirm verfolgen, aber einige Wörter werden am rechten Rand einfach »auseinandergerissen«. Diesem Manko setzt nun unsere kurze Routine »Wordwrap« ein Ende – und das, ohne beim Programmieren lange »herumzutüfteln«. Die kurze Maschinenroutine läuft auf allen XL- und XE-Modellen. Der auszugebende Text muß zunächst einmal in einem maximal 255 Zeichen langen String stehen. Ein String mit dem Namen TEXT\$ soll nun auf dem Bildschirm dargesteilt werden. Der Aufruf der »Wordwrap«Routine erfolgt dann mit: »X=USR(Adresse, ADR(TEXT\$), LEN(TEXT\$))«. Der Wert »Adresse« bezieht sich dabei auf die Anfangsadresse der »Wordwrap«-Routine im Speicher (sie wird in einem String abgefegt).

Weiterhin kann man noch den rechten Rand mit »POKE 83,X« und den linken Rand mit »POKE 82,X« festlegen. Allerdings darf eine Zeite dann nicht kürzer als das längste verwendete Wort sein. (Bastian Robin/wb)

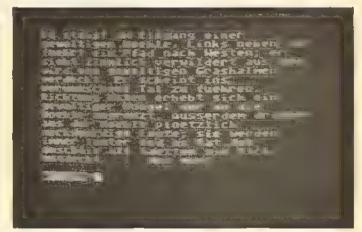
PROGRAMM-STECKBRIEF		
Programmname	Wortumbruch	
Programmtyp	Unlity	
Programmiersprache	Maschinensprache	
Programmlänge	1701 Byte	
für Computer	800 XL/130 XE	
zusätzliche Hardware	keine	
Eingabehille	Prüfsummer	
Bemerkung	Bei Textausgabe auf dem Bildschirm immer zusammenhangende Worte in jeder Zeile.	
Leserservice	Diskette (WORDWRAPBAS)	

Links peben if  I man desten de la
--

Hier sind die Texte am rechten Rand zerstückelt. Es fälft einem schwer, den Text zu lesen, da man am Ende einer Zeile den Rest des Wortes in der nächsten Zeile suchen muß.

50 REM ********************	<ic></ic>
51 REM * Demo Programm fuer den *	(RD>
52 REM * Wordprozessor. Eingabeteil *	<tt></tt>
53 REM *******************	<11>
100 DIM T# (255)	(EE)
105 OPEN #1,12,0,"K:"	<b><bz></bz></b>
106 POKE 62,10:PCKE 83,30	<wb></wb>
110 RESTORE :? "(ESC CTL ()":POSITION 5,	
6:? "A) Text_ohne_RB_Wordprozessor":POSI	
TION 5,10:? "B) Text_mit_RB_Wordprozesso	
r*	<zw></zw>
120 GET #1,A:IF A=66 THEN 200	(EK)
138 IF A<>65 THEN 120	(EL)
140 7 "(ESC CTL ()":7 :FOR W=1 TO 9	<1H>
150 READ Y\$:? T\$;:NEXT W	<cm></cm>
160 ? :? "(ESC CTL =) RETURN";: INPUT T\$:6	
OTO 110	<km></km>
200 ? "(ESC CTL <)":? :FOR W=1 TO 9	<ia></ia>
210 READ T#: GOSUB 1000: NEXT W	<xs></xs>
220 GOTO 160	<nz></nz>
990 REM *************************	<mu></mu>
991 REM * Das folgende 6502 Unter- *	<ma>&gt;</ma>
992 REM * programm druckt den Text- *	<0E>
993 REM * string ohne Zerreissen *	<sy></sy>
994 REM * von Woertern Zeile fuer *	<8K>
995 REM * Zeile auf den Bildschirm. * 996 REM * Laenge und Adresse des *	<rz></rz>
775 REN * Caenge and Hurwase des *	<dv></dv>
777 RED * Strings werden an das * 798 REN * Unterprogramm webergeben. *	<ix></ix>
999 REM **********************	<nv></nv>
1000 A-ADR(T\$):L=LEN(T\$)	<he></he>
1848 X=USR (ADR ("hh (CTL E) Lh (CTL E) KhhpG(	317007
CTL ENHICCTL ANCETE MINE CETE BACCTE , TC	
TL DONIKI P (CTL M) SQ fN \$NDMp+Pm" (ESC CTL	
>>hHDMp(CTL F)1KI_Pt(CTL J)(CTL X)eUES(	
CTL P) (CTL H) -GD_\$qfN\$NDMp (CTL E) 1K(CTL	
X) (CTL P) J) (CTL ,) (CTL M)p(CTL B) (CTL .)	
"),A <sub>1</sub> L)	<bj></bj>
1050 RETURN	<pi></pi>
1990 REM **********************	<rp></rp>
1991 REM * Data-Zeilen fuer *	<0U>
1992 REM # Adventure - Text #	<6J>
1993 REM ***********************************	<sb></sb>
2000 DATA_Du_stehst_am_Eingang_einer_gew	
altigen-Moehle.aLinksanebenadirafuehrtae inaPfadanachaWesten;aerasieht	<nw></nw>
2010 DATA, aziemlich verwildert aus und a	/1460 X
wird_von_gewaltigen_Grashalmen_verdeckt-	
_Er_scheint_ins_verwunschene_Tal	<dw></dw>
2020 DATAzu_fuehrenRichtung_Dsten_e	
rhebt.sich_ein_gewaltiger_Huegel_der_an_	
die_Hoehle_angrenztAusserdem	<ka></ka>
2030 DATAadringen_aus_der_Hochle_ploet	
zlich_merkwuerdige_Laute;_sie_werdeh_imm	
eralauteraundaesahataden	<ez></ez>
2040 DATA_,_Anschein_als_haetten_sie_Dic	41 -4 /
haentdecktaaaWasanun?	<hy></hy>

Listing zu »Wordwrap«



Mit der Routine »Wordwrap« werden Worte am rechten Rand nicht abgeschnitten. Man kann den Text flüssig lesen. Nebenbei wirkt diese Darstellung auch viel professioneller.



## Seitensprung ins Betriebssystem

Einige brauchbare Betriebssystem-Routinen lassen sich auch von Basic aus nutzen. Wir zeigen Ihnen wie.

er Atari-Computer enthält insgesamt 24 KByte ROM, von denen das Basic genau ein Drittel, also 8 KByte einnimmt. Den anderen Teil beansprucht das Betriebssysten.

Dieses enthält alle wichtigen Unterprogramme, die der Computer schon nach dem Einschalten benötigt. So ist beispielsweise Basic zu aktivieren, sind Peripheriegeräte anzusteuern und ähnliches mehr. Das Basic selbst verwendet ständig Routinen des Betriebssystems – dies ist auch der Grund dafür, warum im Basic-Interpreter so viele Befehle zur Verfügung stehen. Es ist beachtlich, daß dafür nur 8 KByte ROM genügen.

Nichts, was mit Ein- und Ausgaben zu tun hat, wie beispielsweise die Befehle PRINT, GRAPHICS, INPUT oder OPEN erledigt das Basic selbst. Es spielen dabei immer Unterprogramme des Betriebssystems mit.

Da es die Atari-Computer bereits seit etwa sechs Jahren gibt und auch verschiedene Versionen (400/800/600XL/800XL/130XE) existieren, wurde das Betriebssystem mehrfach geändert. Wie soll man nun wissen, an welcher Stelle ein bestimmtes Unterprogramm zu finden ist?

Atari hat zum Zeitpunkt der Entwicklung des Betriebssystems vorausgeplant. So wurde dem Atari-Computer eine sogenannte »Sprungleiste« mit auf den Weg gegeben. Das bedeutet, daß man Unterprogramme nicht an ihrer eigentlichen Position im Speicher aufruft, sondern an einer festen Stelle in einer Tabelle von Einsprungvektoren. die sich garantiert nicht ändert. Ein Beispiel: Ob auf dem Atari 400 oder dem Atari 130 XE, mit einem Sprung nach Adresse \$E477 (siehe Tabelle) wird ein Kaltstart ausgeführt. Bei \$E477 selbst steht nur ein Sprungbefehl zur eigentlichen Kaltstart-Routine. Wo sich diese befindet, muß der Programmierer nicht unbedingt wissen.

Dieser Aufbau hat noch einen großen Vorteil: Wer sich nämlich ein anderes Betriebssystem-ROM in seinen Computer einbauen läßt oder sich ein modifiziertes Betriebssystem in den RAM- Bereich von \$C000 bis \$FFFF lädt, kann davon ausgehen, daß alle Programme, die sich an die Sprungtabelle halten, auch auf Ihrer Betriebssystemversion laufen. So kann man mit einem veränderten Betriebssystem beispielsweise einen RAM-Disk-Treiber für den 130 XE einbauen, der dann nicht nur unter DOS 2.5, sondern mit allen Programmen mit Diskettenzugriff funktioniert. Ein solcher Treiber für den zusatzlichen Speicher ist in verschiedenen Spielen bereits integriert.

Betrachten wir die Tabelle der Einsprungadressen, die auf den ersten Blick ziemlich kurz aussieht. Dies lieut ausschließlich daran, daß einige der Routinen fast universell einsetzbar sind. Die Sprungleiste beginnt bei der Adresse 58448 (\$E450), und jede Einsprungstelle belegt natürlich 3 Byte (nämlich 1 Byte für einen Maschinensprache-»JMP« (Sprungbefehl) und 2 Byte für die Adresse). Jede Einsprungadresse hat eine feste Bezeichnung, die von Atari in den entsprechenden technischen Dokumentationen vorgegeben wurde. Ideal wäre es, wenn sich jeder an diese Benennungen halten würde. Dann wäre eine einheitliche Namensgebung gewährleistet andere Programmierer könnten sich schneller in fremde Programme hineindenken. Jeder Name eines Einsprungs hat als letzten Buchstaben ein »v«, als Erinnerung daran, daß es sich ia nur um einen Vektor handelt, und nicht etwa um die tatsächliche Adresse der Routine im Betriebssystem-ROM.

## Fünf Sprünge in die Tiefe

Der zweite Teil der Liste enthält fünf Einsprünge, die man nur auf dem 600 XL, 800 XL und 130 XE nutzen sollte. Programme, die auf dem Atari 400 und 800 laufen sollen, dürfen diese Einsprünge auf keinen Fall verwenden! Man sollte sich also folgendes merken: Programme, die für alle Geräteversionen bestimmt sind, dürfen nur Einsprungadressen zwischen \$E450 und \$E47F benutzen. Alle anderen Aufrufe von Routinen im Bereich ab \$C000 aufwärts sind illegal.

Nach diesen Vorüberlegungen kommen wir zu den einzelnen Routinen. Hier muß man einerseits zwischen den Routinen unterscheiden, die für den Basic-Programmierer interessant sind und andererseits zwischen denen, die nur auf Maschinenspracheebene von Bedeutung sind. Beginnen wir mit einer kurzen Übersicht über die einzelnen Funktionen

58448 \$E450 - DSKIV

Über diese Routine kann man den seriellen Bus für die Benutzung der Diskettenstation initialisieren.

58451 \$E453 - DSKINV

Ein- und Ausgaberoutine für die Diskettenstation, über die man einzelne Diskettenoperationen ausführen kann (beispielsweise Sektoren lesen oder schreiben).

58454 \$E456 - CIOV

Dies ist der Einsprung für das wichtigste Unterprogramm des Betriebssystems, der ClO. Über die ClO (Central Input/Output Utility) können alle standardmäßigen Ein- und Ausgabeoperationen durchgeführt werden (Datei öffnen, Bytes schreiben, Bytes lesen etc.). Bis auf zwei Befehle nämlich, BPUT und BGET (siehe auch den Artikel über eine Schnell-Laderoutine in diesem Sonderheft), sind alle ClO-Funktionen auch über Basic-Befehle ansprechbar.

58457 \$E459 ~ SIOV

Einsprung in die Ein- und Ausgaberoutine für den seriellen Bus. Mit dieser
Routine kann man alle Ein- und Ausgaben über den seriellen Bus durchführen, unter anderem natürlich auch die
unter DSKINV aufrufbaren Funktionen.
58460 \$E45F ~ SETVBV

Diese Routine kann Vektoren von wichtigen Betriebssystemroutinen ändem. Dafür ist eine eigene Routine nötig. Wird nämlich ein Vektor in dem Moment aufgerufen, in dem man ihn verändert, kann dies einen Systemabsturz zur Folge haben. Um einen Vektor zu ändern, muß man zunächst das Xund das Y-Register (High- und Low-Byte) mit dem gewünschten Wert laden. Mit dem Akku legt man dann den zu verändernden Vektor beziehungsweise Zahler fest:

Wert	Name	Bedeutung
0	VIMIRQ CDTMVI-5	IRQ-Vektor Systemzähler 1–5
6	VVBLKI	Vektor auf Immediate-VBI
7	VVBLKD	Vektor auf Deferred-VBI

58463 \$E45F - SYSVBV

Routine zur Beendigung des Immediate-VBI (Vertical Blank Interrupt).



58466 \$E462 - XITVBV

Routine zur Beendigung des Deferred-VBI.

58469 \$E465 - SIOINV

Routine zur Initialisierung des Pokey-Chips, der beim Einschalten nicht initialisiert wird.

58472 \$E468 - SENDEV

Pokey-Chip auf Ausgabe stellen.

58475 \$E468 - INITIV

NMI-Interrupts initialisieren.

58478 \$E46E - CIOINV

Routine zur Initialisierung der CIO (siehe oben).

58481 \$E471 - BLKBDV

Vektor für den Basic-Befehl BYE, der beim Atari 400/800 in den Memo-Pad-Modus und bei den neueren Geräten in den Selbsttest führt.

58484 \$E474 - WARMSV

Vektor für den Warmstart; entspricht dem Drücken der RESET-Taste.

58487 \$E477 - COLDSV

Vektor für den Kaltstart; entspricht dem Aus- und Einschalten des Computers (allerdings wird der zusätzliche RAM-Bereich des XE nicht gelöscht). 58490 \$E47A – RBLOKV

Einen Block vom Kassetten-Recorder einlesen.

58493 \$E47D - CSOPIV

Einen Kanal für das Lesen von Daten vom Kassetten-Recorder öffnen.

58496 \$E480 - PUPDIV

Sprung zum Einschaltbild, das beim XL und XE dem Selbsttest entspricht.

58499 \$E483 - SLFTSV

Sprung in den Selbsttest. 58502 \$E486 -- PHENTV

Eintrag in der Gerätetreibertabelle vornehmen.

58505 \$E489 - PHULNV und 58508 \$E48C - PHINIV

Spezielle Routinen für die Benutzung des parallelen Busses.

Die Erfahrung zeigt, daß von den aufgeführten Routinen nur acht häufiger benutzt werden. Dies sind natürlich zunächst einmal die Einsprünge für Warmstart (von Basic: USR(58484)) und Kaltstart (USR(58487)). Dazu kommen SETVBV, SYSVBV und XITVBV, die bei Interrupt-Routinen des öfteren notwendig sind. Es bleiben die drei Einund Ausgaberoutinen DSK/NV, CIOV und SIOV.

Da alle CIO-Funktionen (bis auf die oben genannte Ausnahme) direkt über Basic-Befehle ansprechbar sind, bleiben also für den Basic-Programmierer noch die beiden Routinen DSKINV und SIOV.

Sie steuern die Ein- und Ausgaben über den seriellen Bus, wobei DSKINV sozusagen eine auf die Diskettenstation »spezialisierte« Version von SIOV ist.

In der obigen Liste war sehr gut zu erkennen, daß man beim Aufruf von

Betriebssystemroutinen die Prozessorregister oft auf bestimmte Werte setzen
muß. Dies ist jedoch mit der USRFunktion nicht möglich. Außerdem kann
man mit USR nur speziell für Basic
geschriebene Routinen aufrufen, da die
mit USR übergebenen Werte automatisch auf dem Prozessor-Stack abgelegt werden. Daher benötigt man eine
spezielle USR-Routine, die zunächst
die Prozessorregister lädt und dann
einen »echten« Sprung ausführt. Ein
solches Programm finden Sie im abgedruckten Basic-Programm.

Bitte speichern Sie das Programm ab, bevor Sie es mit RUN ausprobieren.

Zunächst wird das 35 Byte lange Maschinenprogramm in die Zeichenkette CALL\$ gelesen. Daraufhin wird zu Demonstrationszwecken, mit Hilfe der Betriebssystemroutine DSKINV, der Bildschirm in Grafikstufe 24 mit dem Inhalt der ersten 60 Sektoren auf der Diskette gefüllt. Hier eine Erklärung der Register, die man setzen muß:

### Trickreiche Adressen

769 \$301 - DUNIT

44-000

Nummer des anzusprechenden Diskettenlaufwerks (1 bis 8).

770 \$302 - DCOMND

Kommando der auszuführenden Funktion. Das 1050-Laufwerk von Atari versteht folgende Kommandos: 33 (\*!\*): Formatieren mit 720 Sektoren

34 (>"<): Formatieren mit 1040 Sekto-

80 (»P«): Sektor schreiben ohne Überprüfung (läuft nur auf dem Atari 800, XL und 130 XE)

82 (»R«): Sektor lesen 83 (»S«): Status überprüfen

87 (»W«): Sektor schreiben mit Überprüfung

772,773 \$304,\$305 - DBUFLO,HI

Zeiger auf die Stelle im Speicher, ab der die Informationen eines Sektors von Diskette gelesen oder geschrieben werden.

778,779 \$30A,\$30B - DAUX1/2

Nummer des anzusprechenden Diskettensektors.

Betrachten wir nun den Rest des Beispielprogramms: In den Zeilen 1160 bis 1180 werden den Registern zunächst Konstanten zugeordnet. Dies ist nicht unbedingt notwendig, trägt aber zur Verständlichkeit des Programms bei. Es folgt in Zeile 1190 das Einschaften der hochauflösenden Grafik, Außerdem wird die Anfangsadresse des Bildspeichers berechnet und in der Variablen SC abgelegt. Bereits in Zeile 1200 werden die ersten Register für die DSKINV-Routine gesetzt: DUNIT enthält den Wert 1, weil das erste Laufwerk benutzt werden soll. DCOMND wird auf Lesen von Sektoren (»R« wie Read) gesetzt. Werte, die über den Bereich von 0 bis 255 hinausgehen, erscheinen beim Atari immer als Doppelbytes, bei denen das zweite Byte das höherwertige ist. Konkret heißt das, daß man den Wert

#### Tabelle der wichtigsten Einsprungvektoren V

Adresse			
Hex.	Dez.	Name	Beschreibung
\$E450	58448	DSKIV	Initialisierung des Laufwerks
\$E453	58451	DSKINV	Ein- und Ausgabe für Diskettenstation
\$E456	58454	CIOV	Central Input/Output-Routine (CIO)
\$E459	58457	SIOV	Serial Input/Output-Routine (SIO)
\$E45C	58460	SETVBV	Interrupt-Vektoren setzen
\$E45F	58463	SYSVBV	Beende Immediate-VBI
\$E462	58466	XITVBV	Beende Deferred-VBI
\$E465	58469	SIOINV	Initialisiere POKEY-Chip
\$E468	58472	SENDEV	Schalte POKEY auf Ausgabe
\$E468	58475	INTINV	Interrupts initialisæren
\$E46E	58478	CIOINV	CIO ınitialisieren
\$E471	58481	BLKBDV	Vektor für »BYE«
\$E474	58484	WARMSV	Warmstart-Vektor
\$E477	58487	COLDSV	Kaltstart-Vektor
\$E47A	58490	RBLOKV -	Record von Kassette lesen

Die folgenden Einsprünge gibt es nur auf dem 600 XL, 800 XL und 130 XE!

\$E480	58496	PUPDIV	Vektor zum Einschaftbild
\$E483	58499	SLFTSV	Vektor zum Selbsttest
\$E486	58502	PHENTY	Eintrag in HATABS vomehmen
\$E489	58505	PHULNV	Eintrag löschen
\$E48C	58508	PHINIV	Parallelen Bus initialisieren

einer solchen Adresse folgendermaßen berechnen kann:

»PEEK(Adresse) + 256\* PEEK (Adresse + 1)«

Analog muß man natürlich vorgehen, wenn man ein solches 16-Bit-Register verändern will. Da nur die Sektoren bis Nummer 60 angesprochen werden sollen, muß das High-Byte der Sektornummer immer den Wert 0 enthalten. Es muß also nur ein einziges Mal, vor dem Eintritt in die Hauptschleife, richtig gesetzt werden. Anschließend noch mit der gleichen Methode DBUFLO/Ht auf den Anfang des Bildspeichers bringen. Die Schleife von 1220 bis 1270 benutzt das Indexregister » als Schleifenzähler, so daß man I als Low-Byte der Sektornummer betrachten und in DAUX1 übertragen kann.

Schließlich folgt noch der Aufruf der Betriebssystemroutine. Da DSKINV keine speziellen Werte in den Prozessorregistern erwartet, wird hier einfach jeweils die 0 als Wert angegeben. Zu guter Letzt muß noch das Doppelbyte DBUFLO/HI um genau 128 erhöht werden, da schließlich jeder Sektor 128 Byte lang ist.

Sie haben sicherlich bemerkt, daß dabei nicht unbedingt die größte Ladegeschwindigkeit erreicht wird. Das liegt daran, daß die Basic-Befehle zwischen den einzelnen Aufrufen von DSKINV immer so lange warten müssen, bis die Diskette wieder eine volle Umdrehung zurückgelegt hat. Erst dann kann der nächste Sektor gelesen werden.

## DPEEKs und DPOKEs mit Turbo-Basic XL

Abhilfe schafft Turbo-Basic XL, das einerseits das Atari-Basic an Geschwindigkeit übertrifft, außerdem für die Veränderung von 16-Bit-Registern die schnelleren (und komfortableren) Befehle DPEEK und DPOKE zur Verfügung stellt.

Ungefähr das gleiche kann man mit der Routine SIOV (\$E459) erledigen. Allerdings gibt es auch einige Unterschiede. Da man nicht auf die Diskettenstation festgelegt ist, sondern auch Drucker oder Kassetten-Recorder ansprechen kann, müssen einige Register zusätzlich initialisiert werden (diese Arbeit nimmt einem normalerweise der Einsprungvektor DSKINV ab):

768 \$300 - DDEVIC

Mit diesem Register wird der anzusprechende Gerätetyp ausgewählt:

49 = Diskettenstation

64 = Drucker

96 = Kassetten-Recorder

771 \$303 - DSTATS

Einerseits muß vor dem Aufruf von

1000 REM	<rw></rw>
1010 REM Betriebssystem-Demo	<8T>
1020 REM von Julian F. Reschke	<ze></ze>
1030 REM (c) Happy Computer	<ey></ey>
1040 REM	<si></si>
1050 DIM CALL\$(35):CALL\$(35)="_"	<fs></fs>
1060 PRUEF=0	<1V>
1070 FOR I=1 TO 35	<16>
1080 READ WERT: PRUEF=PRUEF+WERT	<jq></jq>
1090 CALL\$(I,I)=CHR\$(WERT)	<al></al>
1100 NEXT I	<er></er>
1110 IF PRUEF<>4907 THEN PRINT CHR\$(125)	
;"Datenfehler!":END .	<00>
1120 REM	<tg></tg>
1130 DATA_104,104,104,133,212,104,104,13	
3,213,104,104,133,214,104,133,215,104,56	
,233	<bd></bd>
1140 DATA 1,170,165,215,233,0,72,138,72,	
165,212,166,213,164,214,96	<03>
1150 REM	<tp></tp>
1160 DUNIT=769: DCOMND=770	<qr></qr>
1170 DBUFL0=772:DBUFHI=773	<nv></nv>
1180 DAUX1=778: DAUX2=779: DSKINV=58451	<xb></xb>
1190 GRAPHICS 24: SETCOLOR 2,0,0:SC=PEEK(	
88) +256*PEEK (89)	<ex></ex>
1200 POKE DUNIT, 1: POKE DCOMND, ASC("R"): P	
OKE DAUX2,0	<nq></nq>
1210 POKE DBUFHI, INT (SC/256): POKE DBUFLO	
,SC-256*PEEK (DBUFHI)	<ik></ik>
1220 FOR I=1 TO 60	<61>
1230 POKE DAUX1,I	<lx></lx>
1240 WERT=USR(ADR(CALL\$),0,0,0,DSKINV)	<lv></lv>
1250 LO=128+PEEK(DBUFLO): IF LO<256 THEN	
POKE DBUFLO,LO:GOTO 1270	<sl></sl>
1260 LO=LO-256: POKE DRUFHI, 1+PEEK (DBUFHI	
):POKE DBUFLO,LO	<jq></jq>
1270 NEXT I	<f0></f0>
1280 GOTO 1280	<80>
Mit diesem Programm können von Basic aus Betriebssystem-Vektoren angesprungen werden. Anschließend kehrt man wieder ins Basic zurü	ck.

angesprungen werden. Anschließend kehrt man wieder ins Basic zurück.

SIOV festgelegt werden, ob es sich um eine Ausgabe (128) oder eine Eingabe (64) handelt. Andererseits enthält DSTATS im Anschluß daran den Statuscode des seriellen Busses (entspricht normalerweise dem Basic-Fehlercode). 774 \$306 – DTIMLO

Auch die Zeit in Sekunden, die die Ausführung eines Kommandos dauern darf, muß festgelegt werden. Für Diskettenoperation kann man eine 7 wählen (Ausnahme ist das Formatieren, für das man den Wert 160 ansetzen sollte). Für Operationen mit einem Kassetten-Recorder oder einem Drucker empfehlen sich Werte um 30.

776,777 \$308,\$309 - DBYTLO/HI

Anzahl der zu übertragenden Bytes. Bei der Diskettenstation normalerweise 128.

Bleibt anzumerken, daß der Befehl P

(Schreiben ohne Überprüfung) auf dem Atari 400/800 voraussetzt, »SIQV« zu benutzen, da bei den alten Betriebssystemversionen dieser Befehl von DSKINV nicht unterstützt wurde.

Die Routine CALL\$ können Sie übrigens auch verwenden, um eigene Maschinenspracheroutinen von Basic aus aufzurufen. Dies ist immer dann notwendig, wenn die betreffenden Routinen nicht als USR-Routinen für Basic konzipiert wurden.

Es zeigt sich also, daß man mit Einsprungadressen ins Betriebssystem der Atari-Computer viele wirklich nützliche Funktionen ausführen kann. Für den Basic-Programmierer öffnen sich somit viele sonst verschlossene Türen. Denn leider ist Basic in so manch einer Hinsicht einfach überfordert.

(Julian F. Reschke/wb)



## PEEKs und POKEs mit List und Tücke

Wer seinem Atari etwas mehr entlocken will, als es die normalen Basic-Befehle zulassen, der sollte sich der zahlreichen Speicherstellen bedienen, die man mit PEEK und POKE erreicht. as Atari-Basic verfügt schon in der Grundversion über einige sehr komfortable Befehle. Wenn man jedoch beispielsweise bewegte Grafik erzeugen möchte, stößt man in Basic schnell auf Grenzen. Wer sich dann nicht recht mit Maschinensprache anfreunden will, kann sich einer Vielzahl von PEEKs und POKEs bedienen, ohne die ein gutes Programm in Basic schon fast nicht mehr auskommt.

Die Beschreibung der einzelnen Adressen geht jeweils nach einem bestimmten Schema vor sich. Zuerst findet man den dezimalen Wert der Speicherstelle vor. Danach folgt der entsprechende hexadezimale Wert, der jedoch nur bei der Programmierung in Maschinensprache interessant ist.

Viele der Player-Missile-Adressen haben die Eigenart, daß sie, je nachdem ob sie gelesen oder beschrieben werunterschiedliche Funktionen wahrnehmen. Bezieht sich in der Tabelle eine Beschreibung auf das Lesen, so wird dies mit (L), beim Schreiben mit (S) gekennzeichnet. Die Größe eines Players oder Missiles wird folgendermaßen angegeben. Wert 0 oder 2 entspricht der normalen Größe. Wert 1 ergibt die doppelte Größe und Wert 3 die vierfache Größe. Ist der Inhalt eines Kollisionsregisters ungleich Null, so hat ein Zusammenstoß zwischen den jeweils angegebenen Playern oder Missiles stattgefunden.

(Wolfgang Czerny/wb)
Tabelle der wichtigsten PEEKs und POKEs

Dezimal- Wert	Hexadezimal- Wert	Label	Beschreibung	
16	\$10	IRQEN	Die Interruptquellen des Pokey können von hier aus gesteuert werden Beispiel »POKE 16,64.POKE 53774,64« setzen die BREAK-Taste bis zum nächsten Graphics-Befehl außer Funktion.	
17	\$17	IRQST	Zeigt an, ob Interrupts des Pokey aktiv geworden sind. Beispiel: »PEEK(17)=0< sagt aus, daß die BREAK-Taste gedrückt wurde.	
18,19,20	\$12,\$13,\$14	CLOCK	Ein 3-Byte-Wert, der alle ½ Sekunde inkremenhert wird. Zeitdifferenzen kann man wie folgt berechnen: »Start=INT((PEEK(18)*65536)+(PEEK(19)*256)+PEEK(20))/50)«. Die STOP Zeit wird ebenso ermittelt. Die Zeitdifferenz ZEIT errechnet sich dann aus STOP minus START in Sekunden. Eine Uhr kann mit diesen Speicherstellen natürlich auch simuliert werden.	
65	\$41	IOSOUNDEN	Bei einem Wert von Nutl wird bei Input/Output-Operationen, also beispielsweise beim Lesen einer Dis- kette, die Tonausgabe unterdrückt. Beispiel, »POKE 65,0« schaltet den Ton ab, »POKE 65,3« schaltet ihn wieder ein. Auf die Sound-Befehle oder das Tastaturklicken hat dies jedoch keinen Einfluß.	
77	\$4D	ATTRACT	Der sogenannte Attract-Modus ist für den Farbwechsel zur Schonung des Bildschirms verantwortlich. Wird für etwa zehn Minuten keine Taste betätigt, so hat ein Zähler den Wert 128 erreicht und schaltet diesen Modus ein. Dies kann verhindert werden, indem man Im Programm regelmäßig den Befehl »POKE 77,0« durchführt und so den Zähler auf Null zurücksetzt. Mit »POKE 77,129« schaltet man den Attract- Modus sofort ein.	
82	\$52	LMARGIN	Linker Rand bei der Textdarstellung. Der Standardwert ist 2 Mit »POKE 82,0« beispielsweise kann der linke Rand auf Spalte 0 eingestellt werden.	
83	\$53	RMARGIN	Rechter Rand bei der Textdarstellung. Der Standardwert in Grafikstufe 0 ist 39. »POKE 83,X« stellt i rechten Rand auf Spalte X.	
84	\$54	ROWCRS	Diese Speicherstelle beinhaltet die aktuelle Zeilenposition des Cursors in Grafikstufe 0	
85,86	\$55,\$56	COLORS	Aktuelle Spaltenposition des Cursors als 2-Byte-Wert in Grafikstufe 0.	
88,89	\$58,\$59	SCRSTART	Dieser 2-Byte-Wert gibt die Adresse des ersten Bildschirmbytes an Man errechnet sie folgenderma- Ben: »ADR = PEEK(88) + 256 * PEEK(89) « Kennt man diese Adresse, so kann man den Bildschirm leicht speichern oder beliebige Werte direkt auf den Bildschirm POKEn.	
93	\$5d	DATCURS	Hier wird das Zeichen abgelegt, das sich unmittelbar unter dem Cursor befindet.	
106	\$6a	RAMTOP	Durch die Verschiebung des RAM-Tops wird es beispielsweise möglich, mehrere Screens gleichzeitig im Speicher zu behalten und sie dann im sogenannten Page-Flipping hin und her zu schalten.	
186,187	\$ba,\$bb		Dieser 2-Byte-Wert enthält die Zeilennummer beim Abbruch eines Programms. Ein solcher Abbruch kann durch eine Programmunterbrechung mit BREAK, ein STOP oder einen Fehler verursacht werden.	
195	<b>\$</b> C3	ERRSAV	Beim Auftreten eines Fehlers wird der Fehlercode hier gespeichert. Dies ist vor allem dann von Nutzen, wenn ein Programmabbruch mit dem TRAP-Befehl verhindert wird, der jeweilige Fehler aber dennoch registriert werden soll. Beispiel: »ERROR=PEEK(195)«	
559	\$22f	DMACNTL	Antic-Kontrolle. Belegt man diese Speicherstelle mit dem Wert Null, so wird der Bildschirm ausgeschaltet. Je nach Grafikstufe bringt dies eine Geschwindigkeitssteigerung bis zu 30 Prozent. Um den Bildschirm wieder einzuschalten, muß man die Speicherstelle mit ihrem ursprünglichen Wert belegen. Beispiel: »WERT=PEEK(559) POKE 559,0« schaltet den Bildschirm aus »POKE 559,WERT« schaltet ihn wieder ein. POKEt man hier den Wert 62, so bedeutet dies einfache Auflösung und beim Wert 46 doppelte Zeilenauflösung der Player und Missiles.	
560,561	\$230,\$231	DLPTR	Dieser 2-Byte-Wert gibt die Antangsadresse der Display List an. »DL PEEK(560)+256°PEEK(561)«.	
580	\$244	COLDSTR	Wenn in dieser Speicherstelle ein Wert größer Null steht, erfolgt beim Betatigen der »SYSTEM RESET« Taste ein Kaltstart. Das heißt, daß gewisse Initialisierungsvorgänge ablaufen und bei angeschlossenem Diskettenlaufwerk neu gebootet wird. Beispiel: »POKE 580,1«.	

Dezimai- Wert	Hexa- dezimal- Wert	Label	Beschreibung	
623	\$26f		Prioritätsregister der Player-Missiles.	
694	\$2b6	XORKEYMSK	POKEt man eine Zahl in diese Speicherstelle, so wird der Zeichencode der Tastatur um den gePOKEten Wert verschoben Beispiel: »POKE 694,128« bewirkt, daß statt der normalen Buchstaben inverse Buchstaben und Zeichen ausgegeben werden.	
702	\$2be	SHIFTLOCK	Hier kann man die Tastaturcodes bis zu einem gewissen Grad beeinflussen. »POKE 702,128< simuliert die Betätigung der CONTROL-Taste. Tippt man also normale Buchstaben ein, so werden statt dessen die entsprechenden Grafikzeichen ausgegeben. »POKE 702,0« schaftet den Kleinschrift-Modus ein, den man auch über die CAPS-Taste erreicht. »POKE 702,64« schaftet zurück in den normalen Schrift Modus. »POKE 702,128« smuliert die Betätigung der CONTROL-Taste. Tippt man also normale Buchstaben ein, so werden statt dessen Grafikzeichen ausgegeben. »POKE 702,255« bewirkt, daß nur noch Tasten akzeptiert werden, die mit numerischen Werten oder Interpunktion belegt sind. Buchstaben können nur über die SHIFT-Taste erzeugt werden.	
703	\$2bf	NUMNXTLIN	Der hier gespeicherte Wert gibt die Zahl der Textzeilen auf dem Bildschirm an. Es werden nur die Werte 0, 4 und 24 akzeptiert. POKEt man zum Beispiel in Grafikstufe 0 den Wert 4, so erhält man ein vierzeiliges Textfenster am Unteren Bildschirmrand, wie man es von verschiedenen Grafikstufen her kennt. Auf diese Weise kann man auch in Grafikstufe 0 ein vom sonstigen Bildschirm unabhängiges Textfenster einschalten. Wie in Grafikstufe 1 und 2 muß der Bildschirm dann mit PRINT #6 angesprochen werden. Dabei empfiehlt es sich, den Cursor mit »POKE 752,1« zumindest vorübergehend auszuschalten.	
704	\$2c0	COLPMO	Farbregister für Player 0 und Missile 0	
705	\$2c1	COLPM1	Farbregister für Player 1 und Missile 1	
706	\$2c2	COLPM2	Farbregister für Player 2 und Missile 2	
707	\$2¢3	COLPM3	Farbregister für Player 3 und Missile 3	
708	\$204	COLOR0	Farbregister 0 Dies entspricht dem Befehl »SETCOLOR 0,Farbe,Helligkeit«. Um den entsprechende POKE-Wert zu erhalten, multipliziert man den Farbwert mit 16 und addiert den Helligkeitswert. Beispfür die Farbe Blau »SETCOLOR 0,9,4« entspricht dem Befehl »POKE 708,9*16+4« oder »POKE 708,148«. In Grafikstufe 1 und 2 bestimmt dieses Register die Farbe der Großbuchstaben.	
709	\$2c5	COLOR1	Farbregister 1. In Grafikstufe 1 und 2 bestimmt dieses Register die Farbe von Kleinbuchstaben. In Stufe 0 und 8 wird hier die Helligkeit der Zeichen eingestellt.	
710	\$2c6	COLOR2	Farbregister 2. In Grafikstufe 1. und 2. bestimmt dieses Register die Farbe inverser Großbuchstaben, Stufe 0. und 8. die Farbe des Hintergrunds.	
711	\$2c7	COLOR3	Farbregister 3. Werden die vier Missiles zu einem Player vereinigt, so wird die Farbe dieses zusätzlichen Players hier festgelegt. In Grafikstufe 1 und 2 steht hier der Farbwert für inverse Kleinbuchstaben.	
712	\$2c8	COLOR4	Hintergrundfarbe	
729	\$2d9	KEYRPDELY	Bestimmt die Zeit vom Tastendruck bis zum Beginn der Repeat Funktion. Der Standardwert ist 40.	
730	\$2da	KEYREP	Gibt die Wiederholungsfrequenz der Repeat-Funktion an. Der Standardwert ist 5.	
731	\$2db	CLICKDISA	Hier kann man das Tastaturklicken ein- und ausschalten, »POKE 731,255« bedeutet Klicken aus, »POKE 731,0« schaltet das Klicken ein.	
732	\$2dc	HELPFLAG	Diese Speicherstelle gibt an, ob die HELP-Taste gedrückt wurde.  8eispiel: »X=PEEK(732) x= 17 bedeutet HELP-Taste gedrückt x= 81 bedeutet SHIFT-HELP gedrückt x=145 bedeutet CONTROL-HELP gedrückt	
752	\$2f0	CRSINH	Mit dieser Speicherstelle läßt sich der Cursor ein- und ausstellen »POKE 752,1« bedeutet Cursor aus, »POKE 752,0« bedeutet Cursor ein	
755	\$2f3	CHARCTRL	Hier kann man die Darstellung des Cursors und des Textes auf dem Bildschirm beeinflussen. Es sind acht verschiedene Einstellungen möglich. Beispiel: »POKE 755,X«  X=0 – Zeichen sichtbar, Cursor unsichtbar, Schrift normal  X=1 – Zeichen unsichtbar, Cursor unsichtbar, Schrift normal  X=2 – Zeichen sichtbar, Cursor sichtbar, Schrift normal  X=3 – Zeichen unsichtbar, Cursor sichtbar, Schrift normal  X=4 – Zeichen sichtbar, Cursor unsichtbar, Schrift auf dem Kopf  X=5 – Zeichen unsichtbar, Cursor unsichtbar, Schrift auf dem Kopf  X=6 – Zeichen sichtbar, Cursor sichtbar, Schrift auf dem Kopf  X=7 – Zeichen unsichtbar, Cursor sichtbar, Schrift auf dem Kopf  POKEt man mit einer Schleife abwechselnd die Werte 0 bis 3 in diese Speicherstelle, so kann man invers dargestellten Text in Grafikstufe 0 zum Blinken bringen.	
756	\$2f4	CHARBASE	Zeichensatzauswahl. Der Atarı hat neben seinem Standardzeichensatz noch einen internationalen Zeichensatz vorrätig, in dem auch die deutschen Umlaute enthalten sind. Dieser internationale Zeichensatz wird durch »POKE 756,204« aktiviert. Mit »POKE 756,224« schaltet man zurück in den normalen Zeichensatz. Der »POKE 756,226« ermöglicht es, in Grafikstufe 1 und 2 Kleinbuchstaben und Grafikzeichen darzustellen. Allerdings muß man sich dann das Leerzeichen selbst definieren, da in diesem Modus das Leerzeichen als Herz dargestellt wird.	



## PEEKS UND POKES

Dezimal-   Wert	Hexa- dezimal- Wert	Label	Beschreibung	
764	\$2fc	KBCODE	Hier wird der Tastaturcode der zuletzt gedrückten Taste gespeichert. Dieser Code entspricht allerdings nicht dem Standard ASCII-Code, sondem stellt einen Atari-spezifischen Code dar.	
1021-1151	\$3fd-\$47f	CASBUF	Kassettenpufferbereich. Bei der Verwendung von Diskettenlaufwerken kann man diesen Bereich zum Speichern von Maschinenunterprogrammen verwenden.	
1536 1791	\$0600-06ff		Page 6 - Dieser Bereich wird in der Regel zum Speichern von Maschinenunterprogrammen verwende	
53248	\$d000		(L) Kollisionsregister Missile 0 mit Spielfeld. (S) Horizontale Position von Player 0.	
53249	\$4001		(L) Kollisionsregister Missile 1 mit Spielfeld. (S) Horizontale Position von Player 1.	
53250	\$d002		(L) Kollisionsregister Missile 2 mit Spielfeld, (S) Horizontale Position von Player 2.	
53251	\$d003		(L) Kollisionsregister Missile 3 mit Spielfeld. (S) Horizontale Position von Player 3.	
53252	\$d004		(L) Kollisionsregister Player 0 mit Spielfeld. (S) Honzontale Position von Missile 0.	
53253	\$d005		(L) Kollisionsregister Player 1 mit Spielfeld, (S) Horizontale Position von Missile 1.	
53254	\$d006		(L) Kollisionsregister Player 2 mit Spielfeld, (S) Hortzontale Position von Missile 2.	
53255	\$d007		(L) Kollisionsregister Player 3 mrt Spielfeld (S) Honzontale Position von Missile 3.	
53256	\$4008		(L) Kolinsionsregister Missile 0 mit einem Player. (S) Größe von Player 0.	
53257	\$d009		(L) Kollislonsregister Missile 1 mit einem Ptayer. (S) Größe von Ptayer 1.	
53258	\$d00a		(L) Kollisionsregister Missile 2 mit einem Player. (S) Größe von Player 2.	
53259	\$d00b		(L) Kollisionsregister Missile 3 mit einem Player. (S) Größe von Player 3.	
53260	\$d00c		(L) Kollisionsregister Player 0 mit anderem Player. (S) Größe von Missile 0,	
53261	\$d00d		(L) Kollisionsregister Player 1 mit anderem Player (S) Größe von Missile 1.	
53262	\$d00e		(L) Kollisionsregister Player 2 mit anderem Player (S) Größe von Missile 2.	
53263	\$d00f		(L) Kollisionsregister Player 3 mit anderem Player. , (S) Größe von Missile 3.	
53277	\$dO1d		(S) Hier kann man die Player und Missile ein- und ausschalten,	
53278	\$d01e		(S) Schreibt man in dieses Register einen beliebigen Wert, so werden alle Kollisionsregister gelöscht.	
53279	\$d01f	CONSOL	Diese Speicherstelle wird zur Abfrage der Sondertasten des Atan, also von OPTION, SELECT und START verwendet. Da die HELP-Taste nur bei den XL-Modellen vorkommt, iat ihre Abfrage an anderer Stelle zu entnehmen. Die Werte, die man aus dieser Speicherstelle erhält, haben folgende Bedeutung X=PEEK(53279)  x=0 - OPTION+SELECT+START gedrückt  x=1 - OPTION+SELECT gedrückt  x=2 - OPTION+START gedrückt  x=3 - OPTION Taste gedrückt  x=4 - SELECT+START gedrückt  x=5 - SELECT Taste gedrückt  x=6 - START Taste gedrückt  x=7 - keine der Sondertasten wurde betätigt  Die gleichzeitige Betätigung von CONTROL oder SHIFT mit einer der Sondertasten liefert keine anderen Werte.	
53770	\$d20a	RANDOM	Diesem Register kann man eine Zufallszahl entnehmen, deren Wert zwischen 0 und 255 liegt. Das ist vor allem beim Arbeiten mit Compilem wichtig, die die »RND«-Funktion nicht übersetzen.	
54016	\$d300	PORTA	Port A der PIA	
54017	\$d301	PORT8	Port B der PIA	
54018	\$d302	PORTACNTL	Datenrichtungsregister für Port A.	
54019	\$4303	PORTBONTL	Datenrichtungsregister für Port B.	
54279	\$d407	PMBASE	Hier muß das High-Byte der Adresse des Player-Missile-Speichers stehen.	

## Dem SOUND-Befehl auf der Spur

Oft wird den Atari-Computern Ihre Soundfähigkeit abgesprochen. Ein paar Tricks entlocken dem Atari aber dennoch sehr wohlklingende Töne.

PROG	PROGRAMM-STECKBRIEF		
Programmname	Musik-Kura		
Programmtyp	Kurs		
Programmiersprache	Atan-Basic		
Programmlänge	insgesamt 33716 Byte		
für Computer	800XL/130XE		
zusätzliche Hardware	Diskettenlaufwerk oder Kassettenrecorder		
Eingabehilte	Prüfsummer		
8emerkung	zusammenhängender Musik-Kurs (SOUND.7 und SOUND.8 sind Demos).		
Leserservice	Diskette (SOUND.12 bis SOUND.8)		

s ist sicher etwas zuviel verlangt, einem Heimcomputer Musikqualitäten einer elektronischen Orgel abzuverlangen. Schließlich wurde er in erster Linie für andere Zwecke entwickelt. So geht seine Verwendung vom Spielecomputer über Grafikanwendungen bis hin zur Dateiverwaltung und Textverarbeitung. Aber es beschäftigen sich eben auch einige »Musikfreaks« mit den Soundeffekten.

In diesem Beitrag dreht sich alles um den SOUND-Befehl. So gilt es hier, dem Atari-Computer in Sachen Tonerzeugung auf die Sprünge zu helfen. Der eingebaute Sound-Baustein »Pokey« soll also voll ausgereizt werden. Dabei werden Tricks angewandt, die bislang nur Insidern bekannt waren.

Noch ein Wort zum Aufbau dieses Beitrags. Sie finden eine ganze Reihe von dokumentierten Listings, die das Grundgerüst des Artikels darstellen. Nur wenn Sie diese Programme vorliegen haben, können Sie sich richtig in die "Sounds-Materie der Atari-Computer einarbeiten. Es empfiehlt sich also, alle hier abgedruckten Programme einzugeben. Wenn Sie ein Diskettenlaufwerk verwenden, sollten Sie die Programme mit den vorgeschlagenen Programmamen speichern. Nach der Abarbeitung des ersten Programms wird dann automatisch das nächste Programm geladen etc. Verwenden Sie einen Kassettenrecorder, müssen Sie die Programme in der abgedruckten Reihenfolge eingeben und auf einer Kassette speichern. Übrigens: Lassen Sie die Musikprogramme nur unter dem normalen Atari-Basic laufen, Turbo-Basic XL arbeitet die Programme viel zu schnell ab.

Der SOUND-Befehl setzt sich aus den vier Parametern »Voice«, »Pitch«, »Distortion« und »Volume« zusammen. Dabei steht Voice für Stimme, Pitch für Tonhöhe, Distortion für Verzerrung und Volume für die Lautstärke. Damit man sich besser und schneller in die Beispielsprogramme einarbeiten kann, werden immer die gleichen Variablen verwendet. Dies trägt außerdem zur Übersichtlichkeit bei. In den folgenden Beispielen wird immer nur die erste Stimme verwendet. Der Ver-

zerrungsfaktor beträgt dabei stets 10. Die Tonwiedergabe erfolgt bei dieser Einstellung verzerrungsfrei.

Falls Sie ein Diskettenlaufwerk verwenden, geben Sie jetzt bitte »RUN "D:SOUND.12« ein, um Listing 1 zu starten. Beim Kassettenrecorder laden Sie bitte das entsprechende Programm.

Innerhalb der Beispielsprogramme können Sie sich mit der OPTION-Taste eine Tonfolge immer wieder anhören. Mit START gelangen Sie zum nächsten Beispiel.

Der Ablauf von Listing 1:

- Der Computer meldet sich zuerst mit einem mittleren C.
- Verändern der Lautstärke, mal langsam und dann wieder schnell.
- nur die Tonhöhe verändern
- Tonhöhe und Lautstärke verändern

Wenden wir uns nun der Musikprogrammierung zu. Insgesamt stehen Pitch-Werte von 0 bis 255 zur Verfügung. Es sind aber nur die Werte interessant, die jeweils einem Ton entsprechen (siehe Tabelle 1).

Noten- nummer	Tonhõhe	Noten- name
1	14	С
2	55	В
3	16	A# a Bb
4	17	A
5	18	G# a Ab
6	19	G
7	21	F# 0, Gb
В	22	F
9	23	E
10	24	D# o. Eb
11	26	D
12	27	C# a.Db
13	29	C
14	31	В
15	33	A# 0.8b
16	35	A
17	37	G# a Ab
18	40	G
19	42	F# o. Gb
20	45	F
21	47	E
22	50	D# a Eb
23	53	D
24	57	C# a Db
25	60	C

_			
	Noten- nummer	Tonhöhe	Noten- name
7	26	64	В
	27	68	A# a. Bb
	28	72	A
	29	76	G# a. Ab
3	30	81	G
	31	85	F# o. Gb
)	32	91	F
	33	96	E
	34	102	D# o. Eb
,	35	108	D
	36	114	C# o. Db
5	37	121	C
	38	128	8
	39	135	A# a. Bb
)	40	144	A
	41	153	G# o. Ab
	42	162	G
	43	173	F# a.Gb
, l	44	182	F
	45	193	E
	46	204	D# a. Eb
	47	217	D
	48	230	C# a Db
b	49	243	С
	50	255	8
			_

Tabelle 1. Tonhöhentabelle

Ein Unterprogramm bestimmt die Tondauer, das heißt, ob zum Beispiel eine ganze, halbe oder Viertelnote gespielt wird. Es bietet sich hier an, einen im Betriebssystem der Atari-Computer eingebauten Counter (Zähler) zu verwenden. Dazu eignet sich die Speicherstelle 540. Der in dieser Adresse enthaltene Zähler zählt jede 1/50-Sekunde einen Schritt zurück. In die Adresse 540 kann also ein Wert zwischen 0 und 255 gePOKEt werden. Nach Beendigung der Warteschleife, also wenn die Adresse den Wert 0 erreicht hat, kann ins Hauptprogramm zurückgesprungen werden. Hier eine Routine, die eine Sekunde lang eine Note spielt: 10 SOUND 1,200,10,10:REM

Tonausgabe einschalten

20 POKE 540,50:REM

Der Ton soll für eine Sekunde gespielt werden

30 IF PEEK(540) < > 0 THEN 30:REM

Wiederhole bis die Speicherstelle 540 den Wert 0 enthält 40 SOUND 1,0,0,0:REM

Ton abstellen.

Da in einem Programm eine solche Warteschleife öfter benötigt wird, bietet es sich natürlich an, für diesen Zweck eine separate Unterroutine zu verwenden. Weiterhin emp-



fiehlt sich eine Übergabevariable einzusetzen, die bestimmt, wie lange ein Ton gespielt werden soll.

Nachdem eine bestimmte Tonsequenz abgearbeitet wurde, müssen gegebenenfalls alle vier Stimmen wieder abgestellt werden. Die folgende Programmzeile erledigt dies für Sie:

60 FOR AUS=0 TO 3:SOUND AUS,0,0,0:NEXT AUS:RETURN

Wenn man jetzt also ein Musikstück in Basic schreiben möchte, kann man die benötigten Notennummern in DATA-Zeilen ablegen. Mit einer READ-DATA-Schleife lassen sich dann die Werte lesen und verarbeiten. Listing 2 demonstriert dieses Verfahren

Wenn Sie ein Diskettenlaufwerk verwenden, wird Listing 2 automatisch geladen und gestartet. Besitzer eines Kassettenrecorders müssen jetzt wieder das entsprechende Programm laden.

## Begleitung mit Dur-Akkorden

Haben Sie bislang nur einfache Belsplele gehört, folgt jetzt mit Listing 3 eine komplexere Melodie.

Es beginnt mit der Dimensionierung der Variablen. Hier werden einerseits LINE\$ und ein Array N sowie die Variablen V0 bis V3 vordefiniert. In der Variablen N sind dann insgesamt 50 Tonhöhen gespeichert. LINE\$ nimmt immer eine Textzeile auf und stellt sie auf dem Bildschirm dar. Sie dient sozusagen dem Dialog. Anschließend erfolgt ein Sprung zu Zeile 100. Die DATA-Werte ab Zeile 100 werden aber zunächst ignoriert, und es wird ein Unterprogramm ab Zeile 21000 aufgerufen.

Dieses Unterprogramm let für das Löschen des Bildschirms, die Hintergrundfarbe, die Schrift und die Fixierung der Bildränder zuständig. Dazu werden folgende POKEs verwendet:

- »POKE 752,1« Cursor aus
- »POKE 82,X« und »POKE 83,X« für das Setzen des linken und rechten Bildschirmrandes
- »POKE 201,X« setzt die Anzahl der Leerzeichen fest, die bel »PRINT,« ausgegeben werden sollen
- ∍POKE 77,0< schaltet den automatischen Farbwechsel

Nachdem das Unterprogramm abgearbeitet ist, springt das Programm zurück zu Zeile 120. Jetzt werden noch die Noten In das Feld N geschrieben.

Eigentlich stellt sich nun die Frage: Warum werden die Tonhöhen-Werte noch zusätzlich in einem Feld abgelegt? Es würde doch genügen, die benötigten Werte direkt aus den DATA-Zeilen zu lesen. Das ist prinzipiell richtig, aber wenn Sie Tabelle 1 von oben nach unten lesen, wird Ihnen auffallen, daß in der mittleren Spalte verschiede Werte fehlen. In der ersten Spalte, mit der Bezeichnung »Noten-Nummer«, finden Sie aber 50 fortlaufende Nummern. Diese Nummern erlauben eine schnellere Berechnung der Akkorde, Dazu muß nur der Grundton des gewünschten Akkordes angegeben werden und schon wird der gesamte Akkord ausgegeben.

Die Unterroutine zur Berechnung der Akkorde beginnt ab Zeile 40 und nennt sich »Akkord-Berechner«. Vor dem Aufruf dieser Routine muß die Notennummer In der Variablen P übergeben werden. Dabei darf P nur Werte zwischen 8 und 50 enthalten. Weiterhin nimmt noch die Variable PO die entsprechenden Tonhöhen aus dem Feld N an. Dann wird der Akkord berechnet und die Tonhöhenwerte in den Variablen P1, P2 und P3 abgelegt. In Zeile 42 werden die vier Stimmen aktiviert. Stimme 0 ist dabei auf eine höhere Lautstärke gesetzt als die restlichen drei Stimmen des Akkords.

Das Unterprogramm ab Zeile 30 ändert lediglich die Tonhöhe und springt daraufhin in die Verzögerungsschleife. In Zeile 70 sind alle vier Stimmen zunächst auf gleiche Lautstärke gesetzt. Diese wird dann gleichmäßig zurückgenommen, bis nichts mehr zu hören ist. Zeile 210 liest nun die Daten für LINE\$, CHORD, P und WAIT. Der Inhalt von LINE\$ wird auf dem Bildschirm ausgegeben und anschließend das Unterprogramm ab Zeile 40 aufgerufen. In dieser Routine steht also in LINE\$ \*Häns-chen klein - ging allein«, die Variable CHORD enthält nun den Wert 49, P den Wert 30 und WAIT den Wert 30.

Der in Listing 3 verwendete Algorithmus ist nicht für jedes Lied geeignet. Eventuell müssen Sie einige Änderungen vornehmen. Schließlich gibt es verschiedene Tempi und nicht nur Dur-Akkorde. Das abgedruckte Programm zeigt ein Beispiel, wie man die entsprechenden Noten für Akkorde findet und so ein Lied komponiert.

Laden Sie jetzt Listing 4. Dieses Programm spielt jeweils eine Tonleiter in Dur, Moll, und C-Dur vor. Begleitend erscheinen die entsprechenden Listing-Zeilen auf dem Bildschirm. So können Sie sich auch ein Bild von der Programmiertechnik machen.

### **GePOKEte Musik**

Mit den ersten sechs Programmen (Listing 1 bis Listing 6) hatten Sie die Gelegenheit, sich einen Eindruck davon zu verschaffen, was der SOUND-Befehl zu leisten vermag. Allerdings gibt es einen noch viel besseren Weg, nämlich von Maschinensprache aus. Dann kann man die Fähigkeiten des Pokey-Bausteins, der für die Tonausgabe zuständig ist, voll ausnutzen; denn für manche Geräuscheffekte ist das Atarl-Basic einfach zu langsam. Sie müssen aber nicht unbedingt auf Maschinensprache zurückgreifen, denn auch der POKE-Befehl kann die Sound-Register direkt ansprechen. Auf diese Art und Weise lassen sich dann noch viel wirkungsvollere Klangvariationen erzeugen. Listing 7 stellt ein solches Klangbeispiel vor. Zur Bedienung brauchen Sie lediglich einen Joystick. Schließen Sie diesen bitte an Joystickport 1 an.

Auf dem Bildschirm erscheinen dann die Zahlen 53760 bis 53768. Sie repräsentieren die Speicherstellen, die für die Tonausgabe zuständig sind. Nach dem Starten des Programms sind alle Adressen auf Null gesetzt. In der ersten Zeile befindet sich noch ein "> "-Zeichen. Bewegen Sie es mit dem Joystick bis zur Zahl 53768, indem Sie den Joystick nach unten auslenken. Mit dem Feuerknopf verändern Sie die Speicherstelle. Dann brauchen Sie den Joystick nur nach vorne oder hinten zu bewegen. Hat die Adresse den gewünschten Wert erreicht, betätigen Sie nochmals die Feuertaste, und das "> "-Zeichen kehrt zur ersten Zeile

Variable	8edeutung	
Vo	Stimme 0	
V1	Stimme 1	
V2	Stimme 2	
V3	Stimme 3	
P	Tonhöhe	
D	Verzerrung	
V	Lautstärke	
stets ein Wert zugewie	Stimmen (V0 bis V3). Diesen Variablen wird esen, der sich dann während des Programm- rt. Sie erhalten folgende Werte.	
VO	0	
V1	1	
V2	2	
V3	3	
P	Tonhöhe, kann einen ganzzahligen Wert zwischen 0 und 255 annehmen	
D	Verzerrung, nimmt Werte zwischen 0 und 14 an	
V	Lautstärke, für diese Variabien sınd Werte zwischen 0 und 15 zugelassen	

**Variablenliste** 



zurück. »POKEn« Sie beispielsweise in die Adresse 53768 den Wert 80, in Adresse 53760 den Wert 10 und in 53761 einen beliebigen Wert. Anschließend wird der Lautsprecher Ihres Fernsehers oder Monitors aktiv.

Die Speicherzelle 53768 ist der Schlüssel zu allem. Belegen Sie diese Speicherzelle auch einmal mit anderen Werten zwischen 0 und 255. Ändern Sie dann noch die Speicherzellen 53760 und 53761. Was ist nun mit 53762 bis einschließlich 53767? Ändern Sie jetzt auch diese Speicherzellen. So können Sie sich auf experimentellem Wege einige gut klingende Geräusche aussuchen.

Bevor Sie die POKE-Befehle in Basic anwenden, müssen Sie wissen, daß der Sound-Prozessor zunächst installiert werden muß. Dazu reicht es aus, am Anfang eines Programms den Befehl »SOUND 0,0,0,0 « auszuführen. Weiterhin sind die Sound-Register des Atari-Computers nur für Zahlenwerte empfänglich. POKEt man beispielsweise in die Adresse 53762 den Wert 10, erhält man nach »PEEK(53762)« nicht mehr denselben Wert. Warum dies so ist, ist für die Tonerzeugung nicht wichtig. Man muß nur wissen, daß man mit dem POKE-Befehl die Geräuscherzeugung beeinflussen kann. Tabelle 2 listet noch die englischen Bezeichnungen für die Sound-Register auf.

Falls Sie ein Diskettenlaufwerk verwenden, wird am Ende automatisch Listing 8 geladen. Es demonstriert, daß der Atari-Computer schon von Basic aus sehr wohlklingende Töne von sich zu geben im Stande ist. Auch Sie können solche Melodien komponieren, vorausgesetzt Sie haben die hier erläuterten Grundlagen verstanden. Aber auch in der Musikprogrammierung gilt das Motto: Übung macht den Meister

Möchten Sie sich am Schluß noch eine wirklich meisterhaft umgesetzte Melodie anhören? Dann geben Sie noch Listing 9 ein. Nachdem Sie das Programm gestartet haben, hören Sie die Melodie »Digi Loo, Digi Lee«. Das Programm wurde uns freundlicherweise von Kemal Ezcan zur Verfügung gestellt;

Register	Bezeichnung		
53760	Voice 0	Frequency	
53761	Voice 0	Control	
53762	Voice 1	Frequency	
53763	Voice 1	Control	
53764	Voice 2	Frequency	
53765	Voice 2	Control	
53766	Voice 3	Frequency	
53767	Voice 3	Control	
53768	Audio	Control	

Tabelle 2. Die Sound-Register

er ist einer der wenigen, die die Soundprogrammierung des Atari wirklich perfekt beherrschen. Vielleicht machen Sie ihm bald Konkurrenz? (Peter Gerstner/wb)

1000 FOR D=0 TO 14:FOSITION 12,13:? "SOU

REM SOUND.1 Einfuehrung	
(c) 1985 by Peter Gerstner 08/10/85	<sa< th=""></sa<>
1 REM	KUR
2 REM	<ur><li><ur><li>UR</li></ur></li></ur>
3 REM	<b><us< b=""></us<></b>
10 V0=0:V1=1:V2=2:V3=3:DIM TYPE\$(40),N(	5
0):GOTO 100	<ed:< td=""></ed:<>
50 POKÉ 540, WAIT	< VY
52 IF PEEK(540)<>0 THEN 52	<0U
54 RETURN	< ML
60 FOR OFF=0 TO 3:SOUND OFF,0,0,0;NEXT (	3
FF: RETURN	< KN
70 LINE=LEN(TYPE\$):FOSITION HOR,VER	KNG
72 FOR ME=1 TO LINE:? TYPE\$(ME,ME)::IF	Γ
YPES(ME,ME)="_" THEN 76	< DL
74 SOUND 0,25,4,6:FOR DECAY-6 TO 0 STEP	
-0.5:SOUND 0,10,0,DECAY:NEXT DECAY	<ax< td=""></ax<>
76 NEXT ME: RETURN	<nz< td=""></nz<>
100 GOSUB 21000: WAIT-60	< VM
200 TYPE\$-"Bitte_den_Begleittext_fuer_Si	3
UND.12":HOR=3:VER=9:GOSÚB 70	<dl.< td=""></dl.<>
210 TYPE="lesen_und_erst_die_START-Tast	E
e_druecken":HOR=1:VER=11:GOSUB 78	<qs< td=""></qs<>
228 TYPE\$="wennaesalmaTextaverlangtawird	1
":HOR=5:VER 13:60SUB 70	CTV
250 GOSUB 20000:GOSUB 21000	< XK
700 TYPE = "SOUND_Befehl_fuer_das_mittler	-
e_'C'":HOR=3:VER 7:GOSUB 70:GOSUB 50	< 0G
720 POSITION 12,13:? "SOUND.0,121,18,8":	
SOUND 0,121,10,8:60SUB 50:60SUB 60	<16
740 LINE=720:GOSUB 22000	<u0< td=""></u0<>
780 TYPE\$="SOUND_agit_variabler_LAUTSTAN	
RKE":HQR=4:VER=7:GQSUB 70	<m0< td=""></m0<>
820 POSITION 12,13:7 "SOUND.0,60,10,0"	KAZ
840 FOR D=0 TO 14:POSITION 12,13:7 "SOU	
D_0,121,10,";D;"_":SOUND V0,121,10,D	<nm< td=""></nm<>
860 WAIT=20:60SUB 50:NEXT D	<dd.< td=""></dd.<>
880 FOR D=13 TO 0 STEP -1: POSITION 12,13	
:? "SOUND_0,121,10,":D;"A":SOUND V0,121	
10.D	KLV
700 WAIT=20:GDSUR 50:NEXT D:WAIT=60:GDSU	
8 50 .	< JA
920 FOR D=0 TO 14:POSITION 12,13:? "SOUN	
D_0,121,10,";D;"_":SOUND V0,121,10,D	<n3< td=""></n3<>
940 WAIT-10:60SUB 50:NEXT D	<00
960 FOR D=13 TO 0 STEP -1:POSITION 12,13	
:? "SOUND_0,121,10,";D;"_":SOUND V0,121	
10,D	<ls< td=""></ls<>
980 WAIT=10:GOSUB 50:NEXT D:WAIT=60:6050	J
	<jg< td=""></jg<>

ND_0,121,10,";D;".":SOUND V0,121,10,D:NE	
дт р	<bi></bi>
1020 FOR D=12 TO 0 STEP -1: POSITION 12,1	
3:? "SOUND_0,121,10,";D;"_":SOUND V0,121	+
,10,D:NEXT D	<ib></ib>
1030 WAIT-60:60SUB 50	<er></er>
1040 FOR D=0 TO 14 STEP 2:POSITION 12,13	
:? "SOUND_0,121,10,";D; ".":SOUND V0,121,	
10,D:NEXT D	<yi></yi>
1060 FOR D=12 TO 0 STEP -2: POSITION 12,1	
3:? "SOUND_0,121,10,";D;"_":SOUND V0,121	
,10,D:NEXT D:GOSU8 50	<0V>
1080 FOR D=14 TO 0 STEP -2: POSITION 12,1	
3:? "SOUND_0,121,10,";D;"_":SOUND V0,121	
,10,D:NEXT D:60SUB 50	<0B>
1100 FOR D=14 TO 0 STEP -1: POSITION 12,1	
3:7 "SOUND_0,121,10,";D;"_":SOUND V0,121	
,10,D:NEXT D:60SUB 50	<0F>
1120 FOR D=14 TO 0 STEP -0.5:POSITION 12	
,13:? "SOUND_0,121,10,"; INT(D); "_":SOUND	
V0,121,10,D:NEXT D:GOSUB 50	<ca></ca>
1140 FOR D=14 TO 0 STEP -0.2: POSITION 12	
,13:2 "SOUND_0,121,10,"; INT(D); "_":SOUND	
* VØ,121,16,D:NEXT D:SOSUB 56	< <b>ZV</b> >
1160 POSITION 12,13:? "SOUND_0,121,10,10	
": SOUND 0,121,10,10: WAIT 20: GOSUR 50	< VM>
1180 POSITION 12,13:? "SOUND_0,121,10,0_	
":60SUB 60:WAIT=20:G0SUB 50	<3V>
1200 FOR ME=1 TO 3:POSITION 12,13:7 "SOU	
ND_0,121,10,10":SOUND 0,121,10,10:WAIT=1	
0:60SUB 50	<np></np>
1220 POSITION 12,13:2 "SOUND_0,121,10,0_	
": GOSUB 60: GOSUB 50: NEXT ME: WAIT 60: GOSU	
8 50	<8P>
1240 FOR ME=1 TO 5:POSITION 12,13:7 "SOU	477415
ND_0,121,10,10°:SOUND 0,121,10,10	<tn></tn>
1260 POSITION 12,13:? "SOUND_0,121,10,0_	
":GDSUB 60:NEXT ME:WAIT-60:GOSUB 50	<mz></mz>
1300 LINE=820: GOSUB 22000	<yp></yp>
1360 TYPE "SOUND mit variabler TONHOEHE	
":HOR=6:VER=7:GOSUB 70	<mr>&gt;</mr>
1400 FOR P=0 TO 255:POSITION 12,13:? "SO	
UND_0,";P;",10,8":SOUND 0,P,10,8:NEXT	
Р	<0Z>
1420 FOR P=255 TO 0 STEP -1:POSITION 12,	
13:? "SOUND_0,";P;",10,8":SOUND 0,P,10	
Listing 1. Einfache Beispiele mit dem SOUND-Befeh	Ditto
mit dem Namen »D:SOUND.12« auf Diskette speiche	rn.

13-7 **SOUND.0, "19-7, "10, BL":SOUND.0, P., 10					
14-06 FOR PM TO ZSS STEP SHPDSTITUN 12, 13:7 *SOURD.0, * F *, 18,0.4**SOUND 8, P,18, 04.	R-NEYT P	SEUN I	ROKETH ROKETH ROKETH ROKETH ROKETH ROKETH		
Since   Sinc	· ·	(LU)			
35.17   SOURDLO, "19"   19"   19   STEP   -5:POSITION 12, 13:7 "SOURDLO, "19"   19"   19   STEP   19. POSITION 12, 15   19"				ZATS	
14-69 FOR P=255 TO 8 STEP T=5FOSTION 12, 815:7* STUMBLOB, "FF"   16, 824.** STUMBLOB   9, 7, 16   18, 184.**   18, 184.*				ZHIZ	
13:7 *SQUIND.8, "pp", "10, BLA.":SQUIND.8, P10   18   18   18   18   18   18   18	8: NEXT P	<dd></dd>	20020 ? ,"{SHIFF =}_Druecke_START_tuer_w		
188 FOR P=8 TO 258 STEP   18:POSITION 12, 13:7 "SOURD.8, "19;" 18, 8 = FOR P=8 TO 258 STEP   18:POSITION 12, 13:7 "SOURD.8, "19;" 18, 8 = FOR P=8 SOURD 8, P1, 10   12: 13:7 "SOURD.8, "19;" 18, 8 = FOR P=8 SOURD 8, P1, 10   13: 13: 13: 13: 13: 13: 13: 13: 13: 13:	1460 FOR P=255 TO 0 STEP -5:POSITION 12,		eiter_{SHIFT =>"	<pw></pw>	
188 FOR P=8 TO 258 STEP   18:POSITION 12, 13:7 "SOURD.8, "19;" 18, 8 = FOR P=8 TO 258 STEP   18:POSITION 12, 13:7 "SOURD.8, "19;" 18, 8 = FOR P=8 SOURD 8, P1, 10   12: 13:7 "SOURD.8, "19;" 18, 8 = FOR P=8 SOURD 8, P1, 10   13: 13: 13: 13: 13: 13: 13: 13: 13: 13:	13:? "SOUND_0.":P:".10.8":SOUND 0.P.10		20030 2 ,"(CTL Z)(CTL R)(CTL R)(CTL R)(C		
1498 FOR P=0 TO 258 STEP 18:POSITION 12, 15:7 "SOUND.8, 19:P", 18, 63.—"SOUND 8, 19:18   CELL R) CCTL R) CCT		2.000			
13:7 *SOUND.B, *;P;**,18,8_**;SOUND 8,P;18   13:8   13:8   10:8	· ·	7267			
CEX   1368 FOR P=238 TO 8 STEP =18:POSITION 12   137:7 "SUND.4," "PF," 16,84." SOUND 8,9,1   1368 FOR P=238 TO 8 STEP =18:POSITION 12   137:7 "SUND.4," "PF," 16,84." SOUND 8,9,84." SOUND 8,9,1   1368 FOR P=238 TO 8 STEP =18:POSITION 12,13:7 "SUND.4," TO 15:READ PIPER VIOLENCE PROPERTY SOUND 8,9,1   1368 FOR P=238 TO 8 STEP =18:POSITION 12,13:7 "SUND.4," "PF," 16,84." SOUND 8,9,1   1368 FOR P=238 TO 16:READ PIPER VIOLENCE PROPERTY SOUND 8,9,1   14,14.   19,21.   2,13:7 "SUND.4,84.   14,14.   12,13.   12,13.   14,14.   19,21.   2,13:7 "SUND.4,84.   14,14.   12,13.   14,14.   19,21.   2,23.   24,24.   2,23.   24,24.   2,23.   24,24.   2,23.   24,24.   2,23.   2,34.   2,34.   3,33.   3,33.   3,34.   3,3					
1508 FOR P-228 TO 8 STEP -18:POSITION 12   0.B.NEXT P	13:? "SOUND_0,";P;",10,0":SOUND 0,P,10		R) (CTL R) (CTL R) (CTL R) (CTL R) (CTL R) (CTL		
ISSNETT P   10   SETEP - 10   POSITION 12   13:7   SOUND 0, P1, 10   SETEP - 10   POSITION 12   15:00 PO	_8: NEXT P	KEK>	L R) (CTL R) (CTL R) (CTL R) (CTL R) (CTL R) (		
13:70 "SQUADL.B" "FPT", 10, 86_11SQUAD 0, P, 1   13:20 POSITION 12, 13:7 "SQUADD.B", 26, 228   15:60 PYPES" "SQUADD.B", 21:70 SQUADD.B", 21:70   13:80 TYPES" "SQUADD.B", 21:70   13:80 TYPES" "SQUADD.B", 21:70   13:80 TYPES" "TOMMORE "AUTHORITOR TO SQUADD BUSINESS (AT 45, 47, 35, 40, 45, 47, 35, 40, 45, 47, 35, 40, 45, 47, 35, 40, 45, 47, 35, 40, 45, 47, 35, 40, 45, 47, 35, 40, 45, 47, 35, 40, 45, 47, 36, 40, 45, 40, 47, 40, 42, 27, 27, 27, 21; 14, 14, 14, 12, 12, 13; 7 "SQUADD.B", 19; "   16:90 NEXT VIHAIT=16:00SUB SGINEXT MEIRES (AND DEPORTED BUSINESS (AND DEPORTED			CTI RECOTI CES	<bm></bm>	
B.B.NEXT P   1528 POSITION 12,15:7 *SOUND_8,8,8,8.2					
1000B 601LINE=14081GOURD 2,000B,0"   COBD   1050B 11NF=14081GOURD 2,000B   706E 755,3:POKE 755,2:PET=0:GOTO 2   0040B   1050B 179E="SUNDDmailt.variable": HOR THE PROBLEM 75:MOKE 755,0:POKE 755,2:PET=0:GOTO 2   0040B   1050B 179E=2:FICULOR 2,9:0:BETCOLOR			·	(200)	
1568 TYPE="SUND.D.a.b.t.variable":HOR=16     1588 TYPE="SUND.D.a.b.t.variable":HOR=16     1588 TYPE="TONNOEME_und_LAUTSTAGREE":HOR     1588 TYPE="TONNOEME_und_LAUTSTAGREE":HOR     1589 TYPE="TONNOEME und_Lautstagrees     1589 TYPE="TONNOEME und_Lautstagrees     1589 TYPE="TONNOEME und_Lautstagrees     1589 TYPE="TONNOEME und_Lautstagrees     1589 TYPE="TONNOEME und_Lautstagree		<mg></mg>	ETURN	<b0></b0>	
1568 TYPE="SUND.D.a.b.t.variable":HOR=16     1588 TYPE="SUND.D.a.b.t.variable":HOR=16     1588 TYPE="TONNOEME_und_LAUTSTAGREE":HOR     1588 TYPE="TONNOEME_und_LAUTSTAGREE":HOR     1589 TYPE="TONNOEME und_Lautstagrees     1589 TYPE="TONNOEME und_Lautstagrees     1589 TYPE="TONNOEME und_Lautstagrees     1589 TYPE="TONNOEME und_Lautstagrees     1589 TYPE="TONNOEME und_Lautstagree	1520 POSITION 12.13:7 "SOUND.0.0.0.0.0."		20050 POKE 755.3:POKE 755.2:PET=0:GOTO 2		
1560 TYPES**SIGNED_***BICTOLOR** (KT)   1580 TYPE\$**TONNOCHE.und.LAUTSTAREKE**IND R**B-VERF-*5ROSUB 78		CORN	·	(NR)	
1988 TYPE="TONNOBHE_und_LAUTSTAERKE"=1MD R=8*VER=7:609UB 78 MAIT=06*G0SUB 58*REST DRE 1648 1669 DATA_68_53_47_45_48_35_31_29_29_31_ 35_48_45_47_45_56_0 1628 FOR ME=1 TO 16*READ P:FOR V=14 TO 6 STEP -2*POSITION 12_13:7 "SOUND.8_1*;Pi* 1648 WEXT V:MAIT=18*SOSUB 58*NEXT ME: RES 1689 FOR ME=1 TO 26*READ P:FOR V=14 TO 6 STEP -2*POSITION 12_13:7 "SOUND.8_1*,Pi* 12_146_2;P3_243 1689 FOR ME=1 TO 26*READ P:FOR V=14 TO 6 STEP -2*POSITION 12_13:7 "SOUND.8_1*,Pi* 18_1*_V*_1*_*:SOUND 8_1*_10_*,V 12_146_2;P3_243 1689 FOR ME=1 TO 26*READ P:FOR V=14 TO 6 STEP -2*POSITION 12_13:7 "SOUND.8_1*,Pi* 18_1*_V*_1*_*:SOUND 8_1*_10_*,V 16*9 WEXT V:MAIT=18*16565UB 56*NEXT ME 1808 FOR ME=1 TO 26*READ P:FOR V=14 TO 6 STEP -2*POSITION 12_13:7 "SOUND.8_1*,Pi* 18_1*_V*_1*_*:SOUND 8_1*_10_*,V 16*9 WEXT V:MAIT=18*16565UB 56*NEXT ME 1808 FOR ME=1 TO 36*READ P:FOR V=14 TO 6 STEP -2*POSITION 12_13:7 "SOUND.8_1*,Pi* 18_1*_V*_1*_*:SOUND 8_1*_10_*,V 16*9 WEXT V:MAIT=18*16565UB 56*NEXT ME 1808 FOR ME=1 TO 36*READ P:FOR V=15 TO 8 STEP -3*POSITION 12_13:7 "SOUND.8_1*_PI* 18_1*_V*_1*_*:SOUND 8_1*_10_*,V 18_2*_2*_2*_2*_2*_2*_2*_2*_2*_2*_2*_2*_2*_		(00)		414147	
1558 TYPE%="TONHOEHE.aund.LAUTSTAGREE":HO   RE-16408-T-760SUB 70:NAIT-06:150SUB 50:NEST   ORE 16408   DATA_68,53,47,45,40,35,31,29,29,31,   1690 DATA_68,53,47,45,40,35,31,29,29,31,   1620 FOR ME=1 TO 16:READ P:FOR V=14 TO 06   STEP -2:PDSITION 12,13:7 "SOUND.0,9";P;"   CAS)   1640 NEXT V:MAIT=19:S0SUB 50:NEXT ME:RES   VRP.   1660 DATA_23,193,193,193,194,   CAS)   1660 DATA_14,15,19,123,29,40,47,60,81,90,   CAS)   1660 DATA_14,15,19,23,29,40,47,60,81,90,   CAS)   1660 DATA_19,19,19, V					
R-B-1VER-7-8GOSUB 70* WALTH-08/150SUB 508 REST ORE 1680 DAFA_68_55_47_45_48_35_31_79_29_31_3 \$1,40_45_47_55_60_1 1620 FOR ME**I TO 16:READ P:FOR V=14 TO 6 5TEP ~21POSITION 12_13:7: "SOUND_8,**je*;" ,10,**;*y***.**:SOUND 8,P,18_V 1640 NEXT V:MAITH-18:GOSUB 50:NEXT METRES TORE 1660 DAFA_243_193_162_121_96_81_96_47_48 ,29_23_19_14_14_19_23_29_48_47_68_81_96_47_48 ,29_23_19_14_14_19_23_29_48_47_68_81_96_47_48 ,29_23_19_14_14_19_23_29_48_47_68_81_96_47_48 ,29_23_19_14_14_19_23_29_48_47_68_81_96_47_48 ,29_23_19_14_14_19_23_29_48_47_68_81_96_47_48 ,29_23_19_14_14_19_23_29_48_47_68_81_96_47_48 ,29_23_19_14_14_19_23_29_48_47_68_81_96_47_48 ,29_23_19_14_14_19_23_29_48_47_68_81_96_47_48 ,29_23_19_14_14_19_23_29_48_47_68_81_96_47_48 ,29_23_18_18_19_19_10_10_10_10_10_10_10_10_10_10_10_10_10_	: VER=5: 60SUB 70	<kt></kt>			
R=8+UFR-7-60SUB 70*MAIT=60FSUSUB 50FREST LORE 16600 DATA_60FS_3,47,45,46,48,35,31,29,29,31, 35,40,45,47,55,40 1620 FOR ME*1 TO 16:READ P:FOR V=14 TO 6 5TEP -21POSITION 12;13:7 "SOUND_8,* F " ,10,*";V;"-":SOUND_8,P;18,V 1640 NEXT V:MAIT=18:GOSUB 50FNEXT ME:RES TORE 1660 DATA_243,193,162,121,96,91,60,47,40 ,29,23,19,14,14,19,23,29,40,47,60,81,96,47,40 ,29,23,19,14,14,19,23,29,40,47,60,81,96,47,40 ,29,23,19,14,14,19,23,29,40,47,60,81,96,47,40 ,29,23,19,14,114,19,23,29,40,47,60,81,96,47,40 ,29,23,19,14,114,19,23,29,40,47,60,81,96,47,40 ,29,23,19,14,114,19,23,29,40,47,60,81,96,47,40 ,29,23,19,14,114,19,23,29,40,47,60,81,96,47,40 ,29,27,29,10,10,10,10,10,10,10,10,10,10,10,10,10,	1580 TYPE\$="TONHOEHE_und_LAUTSTAERKE":HO		2,2:POKE 83,39:POKE 201.8	<in></in>	
The State					
1600 DAFA_68,55,47,45,48,755,51,29,29,31, 35,48,45,47,55,40 1620 FOR ME=1 TO 16:READ P:FOR V=14 TO 8 1679 FOR ME=1 TO 16:READ P:FOR V=14 TO 8 1679 FOR ME=1 TO 16:READ P:FOR V=14 TO 8 1679 FOR ME=1 TO 16:READ P:FOR V=14 TO 8 1679 DAFA_723,193,162,121,96,91,69,47,40 29,23,19,14,14,19,23,29,40,47,60,81,96,47,40 29,23,19,14,14,19,23,29,40,47,60,81,96,47,40 29,23,19,14,14,19,23,29,40,47,60,81,96,47,40 29,23,19,14,14,19,23,29,40,47,60,81,96,47,40 29,23,19,14,14,19,23,29,40,47,60,81,96,47,40 29,23,19,14,14,19,23,29,40,47,60,81,96,47,40 29,23,19,14,14,19,19,23,29,40,47,60,81,96,47,40 29,23,19,14,14,19,19,23,29,40,47,60,81,96,47,40 29,20,27,29,23,33,35,35,37,40,42,48,47,50,53,5 7,60,64,69,72,76,81,65,91,96 2100 DAFA_14,15,16,17,18,19,21,22,23,24,62,72,93,13,35,35,74,40,42,48,47,50,53,5 7,60,64,69,72,76,81,65,91,96 2100 DAFA_14,15,16,17,18,19,21,22,23,24,62,72,79,31,33,35,77,40,42,48,47,50,53,5 7,60,64,69,72,76,81,65,91,96 2100 DAFA_120,108,114,121,128,136,144,15 3,162,173,162,193,204,217,239,243,255 2120 GOSUB 21100 FOR X=1 TO 50:READ IT:N (X)=TINEXT X  XMD 2100 TERP PER SOURD RESEAD SIDE SIDE TO 8 2100 TERP SIDE SIDE SIDE SIDE SIDE SIDE SIDE SIDE		ZHYS	·		
16.28 FOR ME=1 TO 16:READ P:FOR V=14 TO 8     16.28 FOR ME=1 TO 16:READ P:FOR V=14 TO 8     16.28 FOR ME=1 TO 16:READ P:FOR V=14 TO 8     16.28 FOR ME=1 TO 16:READ P:FOR V=14 TO 8     16.28 FOR ME=1 TO 16:READ P:FOR V=14 TO 8     16.28 FOR ME=1 TO 26:READ P:FOR V=14 TO 8     16.29 FOR ME=1 TO 26:READ P:FOR V=14 TO 8     16.29 FOR ME=1 TO 26:READ P:FOR V=14 TO 8     16.29 FOR ME=1 TO 26:READ P:FOR V=14 TO 8     16.29 FOR ME=1 TO 26:READ P:FOR V=14 TO 8     16.29 FOR ME=1 TO 26:READ P:FOR V=14 TO 8     16.29 FOR ME=1 TO 26:READ P:FOR V=14 TO 8     16.29 FOR ME=1 TO 26:READ P:FOR V=14 TO 8     16.29 FOR ME=1 TO 26:READ P:FOR V=14 TO 8     16.29 FOR ME=1 TO 26:READ P:FOR V=14 TO 8     16.29 FOR ME=1 TO 26:READ P:FOR V=14 TO 8     16.29 FOR ME=1 TO 26:READ P:FOR V=14 TO 8     16.29 FOR ME=1 TO 26:READ P:FOR V=14 TO 8     16.29 FOR ME=1 TO 26:READ P:FOR V=14 TO 8     16.29 FOR ME=1 TO 26:READ P:FOR V=14 TO 8     16.29 FOR ME=1 TO 26:READ P:FOR V=14 TO 8     16.29 FOR ME=1 TO 26:READ P:FOR V=14 TO 8     16.29 FOR ME=1 TO 26:READ P:FOR V=16 TO 8     16.29 FOR ME=1 TO 26:READ P:FOR V=16 TO 8     16.20 FOR ME=1 TO 26:READ P:FOR V=16 TO 8     16.20 FOR ME=1 TO 26:READ P:FOR V=16 TO 8     16.20 FOR ME=1 TO 26:READ P:FOR V=16 TO 8     16.20 FOR ME=1 TO 26:READ P:FOR V=16 TO 8     16.20 FOR ME=1 TO 26:READ P:FOR V=16 TO 8     16.20 FOR ME=1 TO 26:READ P:FOR V=16 TO 8     16.20 FOR ME=1 TO 26:READ P:FOR V=16 TO 8     16.20 FOR ME=1 TO 26:READ P:FOR V=16 TO 8     16.20 FOR ME=1 TO 26:READ P:FOR V=16 TO 8     16.20 FOR ME=1 TO 8		(UA)			
10.20 FOR MEM  TO 16.READ P.FOR V=14 TO 0					
STEP -2:POSITION 12,13:7 "SOUND_8," P "   (AS)   104% NEXT V:MAIT=18:GOSUB 58:NEXT ME:RES YORE 160% 166% NEXT V:MAIT=18:GOSUB 58:NEXT ME:RES YORE 160% 166% DATA_243,193,162,121,94,61,68,47,48   29,23:191,14,14,191,23:29,40,47,68,81,94   (BR)   (CTL R) CCTL R)	35,40,45,47,53,60	<ee></ee>	R3 (CTL R3 (CTL R) (CTL E)"	<jh></jh>	
STEP -2:POSITION 12,13:7 "SOUND.8"," P;"   10," V;" "150;NVP;" "150;NVP;" "150;NVP;" "150;NVP;"   150;NVP;"   15	1620 FOR ME=1 TO 16: READ P: FOR V=14 TO B		21020 ? ."(SHIFT =)_Der_SOUND_Befehl_(SH		
103				<iv></iv>	
1448 NEXT VINAIT=18:50SUB 58:NEXT MEIRES   YR>   1068 1648   YR>   1669   1669   ATA-243,193,162,121,96,81,66,47,48   YR>   1668 FOR ME=1 TO 26:READ P:FOR V=14 TO 8   STEP -2:POSITION 12,131? "SOUND.8,";P;"   CC)   1698 NEXT VINAIT=18:50SUB 58:NEXT ME   CC)   1698 NEXT VINAIT=18:50SUB 58:NEXT ME   CD)   1698 NEXT VINAIT=18:50SUB 57:NEXT VINAIT=18:50SUB 57:NEX	1 1 1 1	(00)			
TORE 1669 1669 DATA_ZA3_193_162_121_96_81_68_47_48 29_23_19_14_14_19_23_29_48_47_68_81_96_ 121_162_193_243 1689 FOR ME=1 TO 26_1READ P.FOR V=14 TO 8 8TEP -2_PDSITION 12_13_17_** 16_9** NEXT V_18A1T=18_1605UB 78:NEXT ME 1708 RESTORE 16_98_1LME=16_28_1605UB 220808 (NX) 20803 REM 20801 REM 20802 REM SOUND_2 BASIC-MUSIK (TL) 20803 REM 20803 RESTORE 21000 2100 DATA_14_15_16_17_18_19_21_22_23_24_4 26_27_27_3_13_3_3_35_37_40_42_45_47_50_35_3_5 7_60_64_68_72_76_61_65_91_96 2110 DATA_102_188_114_122_112_6_136_144_15 3_162_173_182_193_204_217_230_243_255 2120 GOSUB 2010006_18_114_122_112_6_136_144_15 3_162_173_182_193_204_217_230_243_255 2120 GOSUB 2010006_18_114_122_112_6_136_144_15 3_162_173_182_193_204_217_230_243_255 2120 GOSUB 2010006_18_114_122_112_6_136_144_15 3_162_173_182_193_204_217_230_243_255 2120 GOSUB 2010006_180_140_140_140_140_140_140_140_140_140_14		(H8)			
16-60 DATA_243_193_162_121_96_81_68_47_48     27_23_19_14_14_19_23_27_48_47_68_81_96_121_162_193_243     1680 FOR ME=1 TO 26:READ P:FOR V=14 TO 8     STEP2POSITION 12_13:7 "SOUND_8," pp: "     10," ty: ":SOUND_8_1," pp: "     2002 REM					
299.23,19,16,14,19,23,29,40,47,69,81,96, 121,162,193,243 1699 FOR MEST TO 26;READ P;FOR V=14 TO 8 STEP -22POSITION 12,13; "SOUND 8,P;B," 10,";V;"_":SOUND 8,P,10,V 1699 NEXT VIMAIT=10:GOSUB 58:NEXT ME 2008 RESTORE 1600:LINE=1620:GOSUB 220000 (NZ) 2008 RESTORE 1600:LINE=1620:GOSUB 220000 (NZ) 2008 RESTORE 1800:LINE=1620:GOSUB 220000 (NZ) 2008 RESTORE 21000 2100 DATA;14,15,16,17,18,19,21,22,23,24,16,27,27,23,33,53,7,40,46,48,7,27,68,185,71,96 2110 DATA;14,15,16,17,18,19,21,22,23,24,16,27,27,23,23,23,23,24,24,26,27,29,31,33,35,37,40,42,45,47,50,33,57,40,46,48,72,76,81,85,71,96 2110 DATA;14,15,16,17,18,19,21,22,23,24,16,27,27,23,24,16,27,27,23,24,24,26,27,29,31,33,35,37,40,42,45,47,50,33,57,40,46,48,72,76,81,85,71,96 2110 DATA;10,10,10,11,	TORE 1660	<yr></yr>	CCTL RECOTL RECOTL RECOTL RECOTL RECOTL		
299.23,19,16,14,19,23,29,40,47,69,81,96, 121,162,193,243 1699 FOR MEST TO 26;READ P;FOR V=14 TO 8 STEP -22POSITION 12,13; "SOUND 8,P;B," 10,";V;"_":SOUND 8,P,10,V 1699 NEXT VIMAIT=10:GOSUB 58:NEXT ME 2008 RESTORE 1600:LINE=1620:GOSUB 220000 (NZ) 2008 RESTORE 1600:LINE=1620:GOSUB 220000 (NZ) 2008 RESTORE 1800:LINE=1620:GOSUB 220000 (NZ) 2008 RESTORE 21000 2100 DATA;14,15,16,17,18,19,21,22,23,24,16,27,27,23,33,53,7,40,46,48,7,27,68,185,71,96 2110 DATA;14,15,16,17,18,19,21,22,23,24,16,27,27,23,23,23,23,24,24,26,27,29,31,33,35,37,40,42,45,47,50,33,57,40,46,48,72,76,81,85,71,96 2110 DATA;14,15,16,17,18,19,21,22,23,24,16,27,27,23,24,16,27,27,23,24,24,26,27,29,31,33,35,37,40,42,45,47,50,33,57,40,46,48,72,76,81,85,71,96 2110 DATA;10,10,10,11,	1669 DATA 243 193 162 121 96 91 69 47 49		R) (CTL R) (CTL R) (CTL C)*	<lh></lh>	
121,162,193,243 1688 FOR MEE! TO 261READ P:FOR V=14 TO 8 STEP -2:POSITION 12,13:? "SOUND.8, ";P;" 10," "1":"SOUND 8,P,19," (CC) 1698 NEXT V:MAIT=10:GOSUB 56:NEXT ME (RJ) 1708 RESTORE 1600:LINE=1620:GOSUB 22000 (NX) 2000 REM (SZ) 2000 REM (S				CAKS	
1680 FOR MEET TO 26:READ P.FOR V=14 TO 8     STEP -2POSITION 12,13-7 "SOUND.8, ";P;"     18, ";V;" ":SOUND 8,P,10,V				SHICK	
STEP -2:POSITION 12,13:7 "SOUND.8,";P;		 	* *		
18, "; Y; "="SOUND 0, P, 10, V	1680 FOR ME=1 TO 26:READ P:FOR V=14 TO 0		4,9,0:SETCOLOR 1,9,12:POKE 752,1:POKE 8		
10, "i," =":SOUND 0, p.10, V   CCC   1690 NEXT VIBALT=10;60SUB 50:NEXT NE   1700 RESTORE 1600:LINE=1620:GOSUB 22000 (NX)   CCT   R) CCT	STEP -2: POSITION 12.13:? "SOUND_0.":P:"		2.2:POKE 83.39:POKE 201.11	<yd></yd>	
1496 NEXT V:MAIT=10:GÓSUB 50:NEXT ME		KEES	21110 2 . " (CTL D) (CTL R) (CTL R) (CTL R) (C		
1708 RESTORE 1600:LINE=1620:GOSUB 22000					
2000 REM				44.400	
2001 REM   2002 REM   SOUND.2   BASIC-MUSIK   XBD   2003 REM   XTD   2004 REM   XTD   2004 REM   XTD   2004 REM   XTD   2005 REM   XTD   2005 RESTORE   2108   XTD   2005 RESTORE   2108   XTD   2005 RESTORE   2108   XTD   2005 RESTORE   2108   XTD   2005 BASIC-MUSIK   XTD   2005 RESTORE   2108   2005 RESTORE   2108   XTD   2005 RESTORE				<m0></m0>	
2002 REM SOUND.2 BASIC-MUSIK (XB) 2003 REM (TL) 2004 REM (TL) 2004 REM (TL) 2005 RESTORE 2100 2010 DATA.14,15,16,17,18,19,21,22,23,24, 26,27,29,31,33,35,37,40,42,45,47,50,53,5 7,60,64,60,72,76,81,85,91,96 2110 DATA.14,12,1128,136,144,15 3,162,173,162,195,204,217,230,243,255 2120 GOSUB 21100:FOR X=1 TO 50:READ IT:N (X)=IT:NEXT X 2130 TYPE\$="Lesen_sie_bitte_im_Textaweit (XB) 2140 TYPE\$="druecken_sie_aerstadie_astart-taste;":HOR=5:VER=13:GOSUB 70 2150 TYPE\$="wenn_aesim_textaverlangtawir d.":HOR=5:VER=13:GOSUB 70 2150 TYPE\$="wenn_aesim_textaverlangtawir d.":HOR=5:VER=13:GOSUB 70 2150 TYPE\$="wenn_aesim_textaverlangtawir d.":HOR=5:VER=13:GOSUB 70 2150 TYPE\$="bean_sie_bitte_im_textaverlangtawir d.":HOR=5:VER=13:GOSUB 70 2150 TYPE\$="bean_sie_bitte_im_textaverlangtawir d.":HOR=5:VER=13:GOSUB 70 2150 TYPE\$="bean_sie_bitte_im_textaverlangtawir d.":HOR=5:VER=13:GOSUB 70 2200 GOSUB 20000:GOSUB 20000 (GOSUB 20000) 21100 (QC) 2330 LIST 2310 QC) 2330 LIST 2310 QC) 2330 FR P=50 TO 1 STEP -1:FOR V=15 TO 0 STEP -3 :SOUND 0,N(P),10,V:NEXT V:NEXT P 2332 IF LINE<2310 THEN 7:LIST 2330 (QC) 2333 FR P=50 TO 1 STEP -1:FOR V=15 TO 0 STEP -3 :SOUND 0,N(P),10,V:NEXT V:NEXT P 2333 FL LINE=2310:GOSUB 2000 (QC) 2355 FL LINE=2310 THEN 2400 (QC) 2500 RUN "D:SOUND.3":REM CASETTEN BENUE TIZE HUESSEN HIER GRAPHICS 0:END ANSIATT R.UT P.SOUND.3":FOR CASETTEN BENUE TIZE HUESSEN HIER GRAPHICS 0:END ANSIATT R.UT P.SOUND.3":FOR CASETTEN BENUE TIZE HUESSEN HIER GRAPHICS 0:END ANSIATT R.UT P.SOUND.3":FOR CASETTEN BENUE TIZE HUESSEN HIER GRAPHICS 0:END ANSIATT R.UT P.SOUND.3":FOR CASETTEN BENUE TIZE HUESSEN HIER GRAPHICS 0:END ANSIATT R.UT P.SOUND.3":FOR CASETTEN BENUE TIZE HUESSEN HIER GRAPHICS 0:END ANSIATT R.UT P.SOUND.3":FOR CASETTEN BENUE TIZE HUESSEN HIER GRAPHICS 0:END ANSIATT R.UT P.SOUND.3":FOR CASETTEN BENUE TIZE HUESSEN HIER GRAPHICS 0:END ANSIATT R.UT P.SOUND.3":FOR CASETTEN BENUE TIZE HUESSEN HIER GRAPHICS 0:END ANSIATT R.UT P.SOUND.3":FOR CASETTEN BENUE TIZE HUESSEN HIER GRAPHICS 0:END ANSIATT R.UT P.SOUND.3":FOR CASETTEN BENUE TIZE HUESSEN H	2000 REM	<sz></sz>	21120 ? ,"(SHIFT =) BASIC MUSIK (SHIFT =		
2002 REM SOUND.2 BASIC-MUSIK	2001 REM	<td></td> <td>&gt;*</td> <td>&lt;13&gt;</td>		>*	<13>
2003 REM 2004 REM 2050 RESTORE 2100 2100 DATA_14,15,16,17,18,19,21,22,23,24, 26,27,29,31,33,35,37,40,42,45,47,50,53,5 7,60,64,68,72,76,81,85,91,96 2110 DATA_102,108,114,121,128,136,144,15 2120 GOSUB 21100:FOR X=1 TO 50:READ IT:N (X)=IT:NEXT X 2130 TYPE\$="Lesen_Sie_abitte_lma_Textameit er":HOR=5:VER=9:80SUB 70 2140 TYPE\$="wenn_aeaim_textameit er":HOR=5:VER=11:GOSUB 70 2150 TYPE\$="lesen_aim_textameit area,":HOR=3:VER=11:GOSUB 70 2100 GOSUB 20000:GOSUB 21100 (H)> 2200 GOSUB 20000:GOSUB 21100 (H)> 2300 LIST 2310 2330 FOR P=1 TO 50:FOR V=15 TO 0 STEP -3:SOUND 0,N(P),10,V;NEXT P 2333 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 STEP -3:SOUND 0,N(P),10,V;NEXT V:NEXT P 2333 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 STEP -3:SOUND 0,N(P),10,V;NEXT V:NEXT P 2400 RUN "D:SOUND,3":REM CASSETTEN BENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTATT RUN"D:SOUND,3":REM CASSETTEN BENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTATT RUN"D:SOUND,3":REM CASSETTEN BENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTATT RUN"D:SOUND,3":REM CASSETTEN BENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTATT RUN"D:SOUND,3":REM CASSETTEN BENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTATT RUN"D:SOUND,3":REM CASSETTEN BENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTATT RUN"D:SOUND,3":REM CASSETTEN BENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTATT RUN"D:SOUND,3":REM CASSETTEN BENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTATT RUN"D:SOUND,3":REM CASSETTEN BENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTATT RUN"D:SOUND,3":REM CASSETTEN BENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTATT RUN"D:SOUND,3":REM CASSETTEN BENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTATT RUN"D:SOUND,3":REM CASSETTEN BENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTATT RUN"D:SOUND,3":REM CASSETTEN BENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTATT RUN"D:SOUND,3":REM CASSETTEN BENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTATT RUN"D:SOUND,3":REM CASSETTEN BENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTATT RUN"D:SOUND,3":REM CASSETTEN BENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTATT RUN"D:SOUND,3":REM CASSETTEN BENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTATT RUN"D:	2002 REM SOUND, 2 RASIC-MUSIK		21139 2 "VCTI 23VCTI PAVCTI PAVCTI PAVCTI PAVC		
2004 REM 2050 RESTORE 2108 2100 DATA.14,15,16,17,18,19,21,22,23,24, 26,27,29,31,33,35,37,40,42,45,47,50,53,5 7,60,46,66,72,76,81,65,91,96 2110 DATA.102,108,114,121,128,136,144,15 3,162,173,182,193,204,217,239,243,255 2120 GOSUB 21100:FOR X=1 TO 50:READ IT:N (X)=IT:NEXT X 2130 TYPE\$="Lesen.Sie.abittealmaTextaweit er":HOR=5:VER=9:80SUB 70 2140 TYPE\$="Lesen.Sie.abittealmaTextaweit d.":HOR=5:VER=9:80SUB 70 2150 TYPE\$="wenn.aesatmaTextaweit d.":HOR=5:VER=13:GOSUB 21100 (X)=IT:NEXT X 2200 GOSUB 21000:FOR V=15 TO 0 STEP -3 :SOUND 0,N(P),10,V:NEXT V:NEXT P 2330 LIST 2310 STEP -3:SOUND 0,N(P),10,V:NEXT V:NEXT P 2330 FOR P=5 TO 1 STEP -1:FOR V=15 TO 0 STEP -3:SOUND 0,N(P),10,V:NEXT V:NEXT P 2333 FUR P=2310 THEN P 2400 2500 RUN "DISOUND.3":REM CASSETTEN BENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTATT RUN"D:SOUND.3" EINGEBN 20010 P, "(CTL R)(CTL R)(CTL R)(CTL R)(CTL R)(CTL 20010 POSITION 2,20:POKE 201,4:POKE 752,1 1 (CTL R)(CTL R)(CTL R)(CTL R)(CTL R)(CTL 20010 P, "(CTL R)(CTL R)(CTL R)(CTL R)(CTL 20010 P, "(CTL R)(CTL R)(CTL R)(CTL 20010 P, "CTL R)(CTL 20010 P, "CTL R)(CTL 20010 P, "CTL R)(CTL 20010 P, "CTL 20010			·		
2050 RESTORE 2100 2100 DATA14,15,16,17,18,17,21,22,23,24, 26,27,29,31,33,35,37,40,42,45,47,50,33,5 7,60,64,68,72,76,81,85,91,96 2110 DATA102,108,114,121,128,136,144,15 3,162,173,182,195,204,217,239,243,255 2120 GOSUB 21100:FOR X=1 TO 50:READ IT:N (X)=T:NEXT X 2130 TYPE*="LesenaSieabittealmaTextaweit er":HOR=5:VER=7:80SUB 70 2140 TYPE*="drueckenaSieaerstadieaSTART- Taste,":HOR=5:VER=1:60SUB 70 2150 TYPE*="wennaesalmaTextaweit er":HOR=5:VER=1:60SUB 70 2150 TYPE*="wennaesalmaTextaweit er":HOR=5:VER=1:60SUB 70 2150 TYPE*="mennaesalmaTextaweit er":HOR=5:VER=1:60SUB 70 2150 TYPE*="lesenaSieaerstadieaSTART- Taste,":HOR=5:VER=1:60SUB 70 2150 TYPE*="mennaesalmaTextaweit er":HOR=5:VER=1:60SUB 70 2150 TYPE*="mennaesalmaTextaweit er":HOR=5:VER=7:60SUB 70 2150 TYPE*="mennaesalmaTextaweit er":HOR=5:VER=7:60SUB 70 2200 GOSUB 20000:157					
2100 DATA_14,15,16,17,18,19,21,22,23,24, 26,27,29,31,33,53,57,40,42,45,47,50,53,5 7,60,46,68,72,76,81,85,91,96 2110 DATA_102,108,114,121,128,136,144,15 2120 GOSUB 21100:FOR X=1 TO 50:READ IT:N (X)=IT:NEXT X 2130 TYPE\$="Lesen_Sie_bitte_imaTextaweit er":HDR=5:VER=9:GOSUB 70 2140 TYPE\$="druecken_Sie_erstadie_START- Taste,":HOR=5:VER=13:GOSUB 70 2150 TYPE\$="druecken_Sie_erstadie_START- Taste,":HOR=5:VER=13:GOSUB 70 2150 TYPE\$="druecken_Sie_erstadie_START- Taste,":HOR=5:VER=13:GOSUB 70 2150 TYPE\$="menn_essim_textaverlangtawird d.":HDR=5:VER=13:GOSUB 70 2150 TYPE\$="10 50:FOR V=15 TO 0 STEP -3 :SOUND 0,NCP),10,VINEXT V:NEXT P 2320 IF LINE<>2320 IF LINE<>2320 TF LINE<>2320 TF LINE<2310 THEN ?:LIST 2330 2330 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 STEP -3:SOUND 0,NCP),10,VINEXT V:NEXT P 2335 IF LINE=2310 THEN ?:LIST 2330 2335 FOR P=50 SOUND 3.70** 2350 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 STEP -3:SOUND 0,NCP),10,VINEXT V:NEXT P 2335 IF LINE=2310 THEN ?:LIST 2330 2335 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 STEP -3:SOUND 0,NCP),10,VINEXT V:NEXT P 2335 IF LINE=2310 THEN ?2000 2350 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 STEP -3:SOUND 0,NCP),10,VINEXT V:NEXT P 23600 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 STEP -3:SOUND 0,NCP),10,VINEXT V:NEXT P 2350 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 STEP -3:SOUND 0,NCP),10,VINEXT V:NEXT P 23600 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 STEP -3:SOUND 0,NCP),10,VINEXT V:NEXT P 23600 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 STEP -3:SOUND 0,NCP),10,VINEXT V:NEXT P 23600 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 STEP -3:SOUND 0,NCP),10,VINEXT V:NEXT P 23600 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 STEP -3:SOUND 0,NCP),10,VINEXT V:NEXT P 23600 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 STEP -3:SOUND 0,NCP),10,VINEXT V:NEXT P 23600 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 STEP -3:SOUND 0,NCP),10,VINEXT V:NEXT P 23600 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 STEP -3:SOUND 0,NCP),10,VINEXT V:NEXT P 23600 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 STEP -3:SOUND 0,NCP),10,VINEXT V:NEXT P 23600 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 STEP -3:SOUND 0,NCP),10,VINEXT			(CTL R)(CTL R)(CTL R)(CTL R)(CTL C)"	<0V>	
2100 DATA_14_15_16_17_18_19_21_22_23_24_ 26_27_29_31_33_35_37_40_42_45_47_50_55_5 7_60_64_66_72_76_81_85_91_96 2110 DATA_102_108_114_121_128_136_144_15 3_162_173_182_195_204_217_230_243_255 2120 GOSUB 21100:FDR X=1 TO 50:READ IT:N (X)=IT:NEXT X 2130 TYPE\$="Lesen_SIs_babitte_imaTextaweit ter":HDR=5:VER=9:60SUB 70 2140 TYPE\$="druecken_SIs_earstadie_afarter taste,":HDR=5:VER=13:60SUB 70 2150 TYPE\$="wenn_assimaTextaverlangtawir d.":HDR=5:VER=13:60SUB 70 2150 TYPE\$="wenn_assimaTextaverlangtawir d.":HDR=5:VER=13:60SUB 70 2200 GOSUB 20000:GOSUB 70 2150 TYPE\$="menn_assimaTextaverlangtawir d.":HDR=5:VER=13:60SUB 70 2200 GOSUB 20000:GOSUB 70 2150 TYPE\$="menn_assimaTextaverlangtawir d.":HDR=5:VER=13:60SUB 70 2200 GOSUB 20000:GOSUB 70 2150 TYPE\$="menn_assimaTextaverlangtawir d.":HDR=5:VER=13:60SUB 70 2200 GOSUB 20000:GOSUB 70 2210 FOR P=1 TO 50:FOR V=15 TO 0 STEP -3 150UND 0,N(P)_10_VINEXT V:NEXT P 2320 IF LINE<2310 THEN ?:LIST 2330 2330 FOR P=50 TO 1 STEP -15 FOR V=15 TO 0 2355 IF LINE=2310 THEN ?:LIST 2330 2330 FOR P=50 TO 1 STEP -15 FOR V=15 TO 0 2350 FOR P=3:SOUND 0,N(P)_10_VINEXT V:NEXT P 2355 IF LINE=2310 THEN ?:LIST 2330 2350 FOR P=50 TO 1 STEP -15 FOR V=15 TO 0 2350 FOR P=50 TO 1 STEP -15 FOR V=15 TO 0 2350 FOR P=50 TO 1 STEP -15 FOR V=15 TO 0 2350 FOR P=50 TO 1 STEP -15 FOR V=15 TO 0 2350 FOR P=50 TO 1 STEP -15 FOR V=15 TO 0 2500 RUN "D:SOUND.3":REM CASSETTEN BENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTAIT RUN'D:SOUND.3":REM CASSETTEN BENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTAIT RUN'D:SOUND.3":REM CASSETTEN BENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTAIT RUN'D:SOUND.3":REM CASSETTEN BENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTAIT RUN'D:SOUND.3":REM CASSETTEN BENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTAIT RUN'D:SOUND.3":REM CASSETTEN BENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTAIT RUN'D:SOUND.3":REM CASSETTEN BENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTAIT RUN'D:SOUND.3":REM CASSETTEN BENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTAIT RUN'D:SOUND.3":REM CASSETTEN BENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTAIT RUN'D:SOUND.	2050 RESTORE 2100	<il></il>	21140 POKE 77,0:RETURN	<an></an>	
26,27,29,31,33,35,37,40,42,45,47,50,53,5 7,60,64,68,72,76,81,85,91,96 2110 DATA_102,108,114,121,128,136,144,15 3,162,173,182,195,204,217,230,243,255 2120 GOSUB 21100:FOR X=1 TO 50:READ IT:N (X)=IT:NEXT X 2130 TYPE\$="LesenaSis_bitte_imaTextaweit er":HDR=5:VER=9:80SUB 70 2140 TYPE\$="druecken_Sis_berstadis_START= Taste,":HDR=5:VER=11:GOSUB 70 2200 GOSUB 20000:GOSUB 21100 CCT 2300 LIST 2310 2310 FOR P=1 TO 50:FOR V=15 TO 0 STEP -3 :SOUND 0,N(P),10,V:NEXT V:NEXT P 2320 IF LINE<2310 THEN ?:LIST 2330 2330 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 STEP -3:SOUND 0,N(P),10,V:NEXT V:NEXT P 2335 IF LINE=2310 THEN 2400 2335 IF LINE=2310 THEN 2400 2340 CINE=2310:GOSUB 220000 2400 LINE=2310:GOSUB 220000 2500 RUM "0):SOUND.3":REM CASSETTEN BENUE TZER MUESSEN HIER GRAPHICS 0:END ANSIATT RUN'D:SOUND.3": EINSEBEN 2200 0 POSITION 2,15:? "(ESC SHIFT DEL)(ESC SH	2100 DATA_14,15,16,17.18.19.21.22.23.24.		•		
7,60,64,68,72,76,81,85,91,96 2110 DATA-102,108,114,121,128,136,144,15 3,162,173,182,195,204,217,230,243,255 2120 GOSUB 21100:FOR X=1 TO 50:READ IT:N (X)=IT:NEXT X 2130 TYPE\$="Lesen_Sie_bitte_im_Textaweit er":HOR=5:VER=9:80SUB 70 2140 TYPE\$="druecken_Sie_estadie_START=Taste,":HOR=5:VER=13:GOSUB 70 2150 TYPE\$="druecken_Sie_estadie_START=Taste,":HOR=5:VER=13:GOSUB 70 2200 GOSUB 20000:GOSUB 70 2150 TYPE\$="druecken_Sie_erstadie_START=Taste,":HOR=5:VER=13:GOSUB 70 2200 GOSUB 200000:GOSUB 2000 2310 FOR P=1 TO 50:FOR V=15 TO 0 STEP -3:SOUND 0,N(P),10,V:NEXT V:NEXT P 2330 FOR P=3:SOUND 0,N(P),10,V:NEXT V:NEXT P 2330 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 3310 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 3310 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 3310 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 2330 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 2330 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 2350 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 2350 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 2350 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 2350 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 2350 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 2400 LINE=2310 THEN 2400 2500 RUN "0:SOUND.3":REM CASSETTEN SENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTATT RUN"D:SOUND.3":REM CASSETTEN SENUE TZER HUESSEN HIER GRAPHICS 0:END ANSTATT RUN"D:SOUND.3":REM CASSETTEN SENUE TZER HUESSEN HIER SACHL R)(CTL R				< 14142.5	
2110 DATA_102,108,114,121,128,136,144,15     3,162,173,182,193,204,217,230,243,255   XID     2120 GOSUB 21100:FOR X=1 TO 50:READ IT:N     XW				(1007)	
3,162,173,182,193,204,217,230,243,255			· ·		
2120 GOSUB 21100:FOR X=1 TO 50:READ IT:N (X) = IT:N (X) = IT:NEXT X (X) = IT			TL RECOTL RECOTL RECOTL RECOTL RECOTL RE		
(X) = IT: NEXT X   2130	3,162,173,182,193,204,217,230,243,255	<xi></xi>	(CTL R)(CTL R)(CTL R)(CTL R)(CTL R)(CTL		
(X) = IT: NEXT X   2130	2120 GOSUB 21100:FOR X=1 TO 50:READ IT:N		RECOTE RECOTE RECOTE RECOTE RECOTE		
2130 TYPE\$="Lesen_Sie_bitte_im_Text_weit er":HOR=5:VER=9:GOSUB 70				< Y7 \	
er":HOR=5:VER=9:80SUB 70				104/	
2140 TYPE\$="druecken_Steachstadie_START-  Taste,":HOR=3: VER=11:GOSUB 70		4.000		******	
Taste,":HOR=3:VER=11:GOSUB 70  2150 TYPEs="wennaesimaTextaverlangtawir d.":HOR=5:VER=13:GOSUB 70  2200 GOSUB 20000:GOSUB 21100  2300 LIST 2310  2310 FOR P=1 TO 50:FOR V=15 TO 0 STEP -3 :SOUND 0,N(P),10,V:NEXT V:NEXT P  2320 IF LINE<2310 THEN ?:LIST 2330  2320 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 STEP -3 STEP -3:SOUND 0,N(P),10,V:NEXT V:NEXT P  2335 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 STEP -3 STEP -3:SOUND 0,N(P),10,V:NEXT V:NEXT P  2355 IF LINE=2310 THEN 2400  2500 RUN "D:SOUND.3":REM CASSETTEN BENUE  TZER MUESSEN HIER GRAPHICS 0:END ANSTATT RUN"D:SOUND.3" EINGEBEN  20000 POSITION 2,20:POKE 201,4:POKE 752, 1 20010 7,"(CTL R)(CTL R)(CTL R)(CTL R)(CTL R)(CTL R)  20010 7,"(CTL Q)(CTL R)(CTL R)(CTL R)(CTL R)(CTL R)  LR)(CTL R)(CTL R)(CTL R)(CTL R)(CTL R)(CTL C)"  22040 F PEEK(53279)=6 THEN POKE 755,2:G  OSUB 21000:RETURN  22050 IF PEEK(53279)=3 THEN POP :POKE 75 5,2:POSITION 2,19:? "(ESC SHIFT DEL)(ESC SHIFT DEL)";:RETU RN  (MZ)  LISTING 1. EINfache Beispiele mit dem SOUND-Befehl.				<uv></uv>	
Taste,":HOR=3:VER=11:GOSUB 70  2150 TYPEs="wennaesimaTextaverlangtawir d.":HOR=5:VER=13:GOSUB 70  2200 GOSUB 20000:GOSUB 21100  2300 LIST 2310  2310 FOR P=1 TO 50:FOR V=15 TO 0 STEP -3 :SOUND 0,N(P),10,V:NEXT V:NEXT P  2320 IF LINE<2310 THEN ?:LIST 2330  2320 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 STEP -3 STEP -3:SOUND 0,N(P),10,V:NEXT V:NEXT P  2335 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 STEP -3 STEP -3:SOUND 0,N(P),10,V:NEXT V:NEXT P  2355 IF LINE=2310 THEN 2400  2500 RUN "D:SOUND.3":REM CASSETTEN BENUE  TZER MUESSEN HIER GRAPHICS 0:END ANSTATT RUN"D:SOUND.3" EINGEBEN  20000 POSITION 2,20:POKE 201,4:POKE 752, 1 20010 7,"(CTL R)(CTL R)(CTL R)(CTL R)(CTL R)(CTL R)  20010 7,"(CTL Q)(CTL R)(CTL R)(CTL R)(CTL R)(CTL R)  LR)(CTL R)(CTL R)(CTL R)(CTL R)(CTL R)(CTL C)"  22040 F PEEK(53279)=6 THEN POKE 755,2:G  OSUB 21000:RETURN  22050 IF PEEK(53279)=3 THEN POP :POKE 75 5,2:POSITION 2,19:? "(ESC SHIFT DEL)(ESC SHIFT DEL)";:RETU RN  (MZ)  LISTING 1. EINfache Beispiele mit dem SOUND-Befehl.	2140 TYPE\$="druecken_Sie_erst_die_START-		22020 7 ,"{SHIFT =}_START=Weiter		
2150 TYPEs="wennaesimaText.verlangt.wir d.":HOR=5:VER=13:GOSUB 70				<je></je>	
d.":HOR=5:VER=13:GOSUB 78					
2200 GOSUB 20000:GOSUB 21100	The state of the s	COTY	·		
2300 LIST 2310 2310 FOR P=1 TO 50:FOR V=15 TO 0 STEP -3 :SOUND 0,N(P),10,V:NEXT V:NEXT P		1			
2310 FOR P=1 TO 50:FOR V=15 TO 0 STEP -3 :SOUND 0,N(P),10,V:NEXT V:NEXT P		<hv></hv>			
2310 FOR P=1 TO 50:FOR V=15 TO 0 STEP -3 :SOUND 0,N(P),10,V:NEXT V:NEXT P	2300 LIST 2310	<00>	RESCOTE RESCOTE RESCOTE RESCOTE RESCOTE		
20010 0,N(P),10,V:NEXT V:NEXT P 2320 IF LINE 2320 IF LINE 2320 IF LINE 2330 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0 2330 FOR P=3:SOUND 0,N(P),10,V:NEXT V:NEXT P (YM) 2335 IF LINE=2310 THEN 2400	2310 FOR P=1 TO 50:FOR V=15 TO @ STEP -3			<z6></z6>	
2320 IF LINE<>2310 THEN ? :LIST 2330				1-07	
2330 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0  STEP -3:SOUND 0,N(P),10,V:NEXT V:NEXT P (YM)  2335 IF LINE=2310 THEN 2400				/ T- 10	
STEP "3:SOUND 0,N(P),10,V:NEXT V:NEXT P (YM)  2335 IF LINE=2310 THEN 2400		CMOS		<tv></tv>	
2335 IF LINE=2310 THEN 2400	2330 FOR P=50 TO 1 STEP -1:FOR V=15 TO 0				
2335 IF LINE=2310 THEN 2400	STEP -3:SOUND 0,N(P),10,V:NEXT V:NEXT P	<ym></ym>	5,2:POSITION 2,19:? "(ESC SHIFT DEL) (ESC		
2400 LINE=2310:60SUB 22000 (YP) 2500 RUN "D:SOUND.3":REM CASSETTEN BENUE TZER MUESSEN HIER GRAPHICS 0:END ANSTATT RUN"D:SOUND.3" EINGEBEN (BX) 20000 POSITION 2,15:7 "(ESC SHIFT DEL)(ESC SHIFT D SC SHIFT DEL)(ESC SHIFT DEL)(ESC SHIFT D L)(ESC SHIFT DEL)(ESC SHIFT DEL)";:RETU RN (MZ: TL R)(CTL R)(CTL R)(CTL R)(CTL R) Listing 1. Einfache Beispiele mit dem SOUND-Befehl.					
2500 RUN "D:SOUND.3":REM CASSETTEN BENUE TZER MUESSEN HIER GRAPHICS 0:END ANSTATT RUN"D:SOUND.3" EINGEBEN CBX 2000 POSITION 2,15:7 "(ESC SHIFT DEL)(ESC SHIFT D CSC SHIFT DEL)(ESC SHIFT D CSC S				Z CHIZ	
TZER MUESSEN HIER GRAPHICS 0:END ANSTATT RUN"D:SOUND.3" EINGEBEN  20000 POSITION 2,15:2 "(ESC SHIFT DEL)(ESC SHIFT DEL)(ESC SHIFT DEL)(ESC SHIFT DEL)(ESC SHIFT DEL)(ESC SHIFT DEL)(ESC SHIFT DEL)";:RETU  20010 7 ."(CTL Q)(CTL R)(CTL R)(CTL R)(CTL R)  Listing 1. Einfache Beispiele mit dem SOUND-Befehl.		416			
RUN"D:SQUND.3" EINGEBEN  20000 POSITION 2,20:POKE 201,4:POKE 752,  1 (PR)  20010 7 ,"(CTL Q)(CTL R)(CTL R)(CTL R)(CTL R)  Listing 1. Einfache Beispiele mit dem SOUND-Befehl.				<xf></xf>	
RUN"D:SOUND.3" EINGEBEN  20000 POSITION 2,20:POKE 201,4:POKE 752,  1	TZER MUESSEN HIER GRAPHICS Ø:END ANSTATT		25000 POSITION 2,15:7 "(ESC SHIFT DEL) (E		
200000 POSITION 2,20:POKE 201,4:POKE 752,  1	RUN"D:SOUND.3" EINGEBEN	<bx></bx>			
1 CPR RN CMZ: 20010 7 . CCTL R) CCTL R) CCTL R) CCTL R) CCTL R)  Listing 1. Einfache Beispiele mit dem SOUND-Befehl.					
20010 7 .* (CTL Q) (CTL R)		(PP)	· · · · · · · · · · · · · · · · · · ·	ZMZN	
TL R) (CTL R) (CTL R) (CTL R) (CTL R) (CTL R) (CTL R) Listing 1. Einfache Beispiele mit dem SOUND-Befehl.			TUP	VIIIZ >	
	*		Lieting & Cinfocks Dejeniete mit dem COURTS Detak		
ACT DAYON DAYON DAYON DAYON (Cablus)	TE RECOTE RECOTE RECOTE RECOTE RECOTE RE			11.	
TOTAL RANGE RANGEL RANGEL RANGEL IT (SCHIED)	(CTL R) (CTL R) (CTL R) (CTL R) (CTL R)	ŀ	(Schluß)		

Ø REM SOUND.3	( 1	N (CHORD-7)	<pv></pv>
c) 1985 by Peter Berstner 10/10/8	B5 <rb></rb>	42 SOUND V0,P0,10,14:SOUND V1,P1,10	,6:50
1 REM	<uq></uq>	UND V2,P2,10,6:SOUND V3,P3,10,6	<fx></fx>
2 REM	<ur></ur>	50 POKE 540, WAIT	<vy></vy>
3 REM '	<us></us>	52 IF PEEK (548) <>0 THEN 52	<8U>>
10 DIM LINE\$(40),N(50):V0=0:V1=1:	:V2=2:V3	54 SOUND V0,0,0,0:RETURN	<6C>
=3:GOTO 100	<tj></tj>		
30 SOUND V0,N(P),10,14:GOTO 50	<y0></y0>	Listing 2. Ein Lied zum Mitsingen. Bitte mit d	em Namen
40 P0=N(P):P1=N(CH0RD):P2=N(CH0R)	D-4):P3-	»D:SOUND.3« auf Diskette speichern.	
		·	



<PV>

<FX> <0U> <6C>

As FER OFF OF TO 3: SOUND OFF, 6.8, 6.9.NEXT OFF SECRED 1108				
FF.RETURN 70 PSBW(1): piew(CMCDD): P2=W(CMCDRD-4): p52 W(CMCDC-7) PSBW(1): piew(CMCDD): P2=W(CMCDRD-4): p52 W(CMCDC-7) PSBW(1): piew(CMCDD): P2=W(CMCDD-4): p52 W(CMCDC-7) PSBW(1): piew(CMCDD): P2=W(CMCD-4): p52 W(CMCDC-7) PSBW(1): piew(CMCDD): p2-X-1-10-10-10-10-10-10-10-10-10-10-10-10-1	60 FOR OFF=0 TO 3: SOUND OFF-0-0-0-NEXT D		A10 FOR L=1 TO 5	<0V>
78 PP=N(C): P1-NNCDORD): P2-NNCDROD-4): P5-NNCDROD-4): P5-NNCDROD-4): P1-NNCDROD-4): P5-NNCDROD-4): P1-NNCDROD-4): P5-NNCDROD-4): P1-NNCDROD-4): P5-NNCDROD-4): P1-NNCDROD-4): P5-NNCDROD-4): P5-NNCDRO	, , ,	21/0/5		
AGE GESSIS 20000  7. 2 FOR DECAY=S TO 9 STEP -1: SCIND VB, PC, PC, PC, PC, PC, PC, PC, PC, PC, PC		ZVVV		
27 FOR DECAYS TO B STEP -1 SUND WB,PB, 18, 10, 2017 SUND WB,PB, 18, 15, 16, 17, 18, 19, 21, 12, 22, 57, 24, 24, 27, 29, 31, 33, 33, 37, 48, 42, 45, 47, 18, 55, 31, 102, 173, 182, 173, 133, 37, 48, 42, 45, 47, 18, 55, 31, 102, 173, 182, 173, 182, 173, 193, 174, 48, 24, 34, 38, 31, 102, 173, 182, 173, 182, 173, 194, 174, 238, 243, 253, 1102, 173, 182, 173, 182, 173, 182, 173, 182, 173, 182, 173, 184, 174, 183, 183, 183, 183, 183, 183, 183, 183	70 P0=N(P):P1=N(CHORD):P2=N(CHORD-4):P3=		630 READ X,Y:LIST X,Y	<dl></dl>
72 FOR DECAY=S TO 8 STEP -1 SUND V8,PQ, 19, 0,000 PATA-8, 19, 0,000 PATA-8, 19, 0,000 PATA-8, 19, 0,000 PATA-8, 19, 19, 12, 12, 22, 23, 24   72, 72, 73, 133, 33, 37, 48, 42, 45, 47, 39, 53, 100, 173, 110, 113, 115, 115, 115, 115, 115, 115, 115	N(CHORD-7)	(PV)	640 GOSUB 20000	<sy></sy>
18. DECAY-SOUND US_P1_18. DECAY-SOUND US_P2_18. DECAY-SOUND US_P1_18. DECAY-SOUND US_P1_				(FE)
2,118, DECAY-SOUND US, P3, 18, DECAY 74 NEXT DECAY RETURN 75 NEAD LIVES, P3, 14, 151, 151, 151, 152, 152, 153, 153, 153, 153, 153, 154, 154, 154, 151, 151, 151, 151, 151				
7A NEXT DECAY:RETURN 120 DATA_14_13_16_13_16_17_18_19_21_22_23_24 120_27_29_31_25_35_37_48_42_46_17_58_55_7 120_DATA_18_18_114_11_16_11_16_11_16_16_16_16_16_16_16_16_	10, DECAY: SOUND VI,P1,10, DECAY: SOUND V2,P		655 FUR L-1 TU 3	CUM2
74 NEXT DECONVECTURE  108 DATA_14_15_16_17_16_17_16_19_21_22_22_24  108 DATA_16_18_17_16_17_16_19_21_22_22_24  108 DATA_16_21_18_11_14_15_16_18_18_24_17_28_16_5  110 DATA_16_21_18_11_14_15_16_11_12_18_18_16_14_155  110 DATA_16_21_18_11_14_11_16_11_12_18_18_16_14_155  120 DECOND 12_188_11_14_17_12_11_18_18_16_14_155  120 DECOND 12_188_11_14_17_12_11_18_18_16_11_111_18_18_18_18_18_18_18_18_18_18_18	2,10,DECAY:SOUND V3.P3.10.DECAY	(XP)	660 READ X,Y:LIST X,Y	<dr></dr>
100 DATALD, 14, 15, 14, 17, 19, 19, 21, 22, 23, 24 76, 27, 27, 29, 13, 33, 33, 33, 37, 49, 42, 45, 47, 49, 55, 57 76, 10, 61, 61, 61, 72, 76, 19, 55, 91, 76 77, 10, 61, 61, 61, 72, 76, 19, 55, 91, 76 77, 10, 61, 61, 61, 72, 76, 19, 55, 91, 76 77, 10, 61, 61, 61, 72, 76, 19, 55, 91, 76 77, 100, 61, 61, 72, 76, 19, 55, 91, 76 77, 100, 61, 61, 72, 76, 19, 55, 91, 76 77, 100, 61, 61, 72, 76, 19, 55, 91, 76 77, 100, 61, 61, 72, 76, 19, 55, 91, 76 77, 100, 61, 61, 72, 76, 19, 55, 91, 76 77, 100, 61, 61, 72, 76, 19, 55, 91, 76 77, 100, 61, 61, 72, 76 77, 100, 61, 61, 72, 76 77, 100, 61, 61, 72, 76 77, 100, 61, 61, 72, 76 77, 100, 61, 61, 76 77, 100, 61, 61, 77, 77 77, 100, 61, 61, 61, 61, 61, 61, 61, 61, 61, 61			The state of the s	<185
79. (29. 31, 23. 33, 37. 34, 32, 34. 32, 39. 55. 57. 69. 34, 68. 72. 76. 68. 72. 76. 68. 72. 76. 68. 72. 76. 68. 72. 76. 68. 72. 76. 68. 72. 76. 68. 72. 76. 68. 72. 76. 68. 72. 76. 68. 72. 76. 78. 78. 78. 78. 78. 78. 78. 78. 78. 78		CXIZ		
11.0 DATA_1.02_108_114_12_128_13_0_14_4_153   1.02_173_182_182_182_182_182_183_14_2   1.03_173_182_182_182_182_183_14_2   1.03_182_182_182_182_183_14_2   1.03_182_182_182_182_182_182_182_182_182_182	100 DATA_0,14,15,16,17,18,19,21,22,23,24			(IE)
11.0 DATA_1.02_108_114_12_128_13_0_14_4_153   1.02_173_182_182_182_182_182_183_14_2   1.03_173_182_182_182_182_183_14_2   1.03_182_182_182_182_183_14_2   1.03_182_182_182_182_182_182_182_182_182_182	.26,27,29,31,33,35,37,40,42,45,47,50,53		700 GRAPHICS 18: SETCOLOR 0,1,10: SETCOLOR	
116 DATALIDS, 1184, 114, 121, 129, 135, 144, 155 162, 173, 128, 173, 128, 173, 128, 173, 128, 156, 144, 155 126 DESUB 210003-FOR X 0 TO SEREAD LITEN CARD 126 DERAD LINES, CHORD, P, MAITE DOSUB 30 IN EXT MC. GOING 60: MAITE DOSUB 30 IN EXT MC. GOI		20V\		< VIDS
1.62_173_182_193_204_217_228_473_255   CPP     1.02 GOSINE 21_48_2 ":2"   CREAD CLIMAS_CALLED TIME (ASSETTED SERVICE		(B1)		1102
128   DOSUB 218082 FOR X S TO SE READ TITME	110 DATA,102,108,114,121,128,136,144,153		710 ? #6:? #6:? #6; "DRUELKE_Option":?	
128   DOSUB 218082 FOR X S TO SE READ TITME	.162.173.182.193.204.217.230.243.255	(ED)	#A: "FUER.WIEDERHOLEN"	<hr/> >
X)=T1.REXIT X   Page		XEE/		
209 POKE 82,412 """"   C91 208 READ LINES, CHORD, P, WAITE? LINES: BOSU 8 49 207 READ LINES, CHORD, P, WAITE SOSUB 30:N EXT ME: GOODE 69: WAITE SOSUB 50:N 209 READ LINES, CHORD, P, WAITE? 12 LINES: G 200 PER REL TU 2: READ P, WAITE SOSUB 30:N EXT ME: GOODE 69: WAITE SOSUB 50:N 200 PER REL TU 2: READ P, WAITE SOSUB 30:N EXT ME: GOODE 69: WAITE SOSUB 50:N 200 PER REL TU 2: READ P, WAITE SOSUB 30:N EXT ME: GOODE 69: WAITE SOSUB 50:N 200 PER REL TU 2: READ P, WAITE SOSUB 30:N EXT ME: GOODE 69: WAITE SOSUB 50:N EXT ME: GOODE 69: WAITE SOSUB	120 GUSUS ZIMMO:FOR X W TU SM:READ IT:NO			KHV2
209 POKE 82,412 """"   C91 208 READ LINES, CHORD, P, WAITE? LINES: BOSU 8 49 207 READ LINES, CHORD, P, WAITE SOSUB 30:N EXT ME: GOODE 69: WAITE SOSUB 50:N 209 READ LINES, CHORD, P, WAITE? 12 LINES: G 200 PER REL TU 2: READ P, WAITE SOSUB 30:N EXT ME: GOODE 69: WAITE SOSUB 50:N 200 PER REL TU 2: READ P, WAITE SOSUB 30:N EXT ME: GOODE 69: WAITE SOSUB 50:N 200 PER REL TU 2: READ P, WAITE SOSUB 30:N EXT ME: GOODE 69: WAITE SOSUB 50:N 200 PER REL TU 2: READ P, WAITE SOSUB 30:N EXT ME: GOODE 69: WAITE SOSUB 50:N EXT ME: GOODE 69: WAITE SOSUB	X)=IT:NEXT X	<ab></ab>	730 ? #6:? #6; "DRUECKE_start": ? #6; "_	
218 READ LINES, CHORD, P, WAITS CLINES 60538 38 10 228 FOR MC=1 TO 2:READ P, WAITS COURS 38 10 229 READ LINES, CHORD, P, WAITS COURS 38 10 230 READ LINES, CHORD, P, WAITS COURS 38 10 230 READ LINES, CHORD, P, WAITS COURS 38 10 230 READ LINES, CHORD, P, WAITS COURS 38 10 230 READ LINES, CHORD, P, WAITS COURS 38 10 231 READ LINES, CHORD, P, WAITS COURS 38 10 232 READ LINES, CHORD, P, WAITS COURS 38 10 233 READ LINES, CHORD, P, WAITS COURS 38 10 234 READ LINES, CHORD, P, WAITS COURS 38 10 235 READ LINES, CHORD, P, WAITS COURS 38 10 236 FOR MC=1 TO 2:READ P, WAITS COURS 38 10 237 READ LINES, CHORD, P, WAITS COURS 38 10 238 READ LINES, CHORD, P, WAITS COURS 38 10 239 READ LINES, CHORD, P, WAITS COURS 38 10 230 FOR MC=1 TO 4:READ P, WAITS COURS 38 10 230 FOR MC=1 TO				<505
28   60   RE-1 TO 2:READ P, MAIT: 50SUB 38:N   CLD 225   READ CURBO, P, MAIT: 50SUB 58   CLD 226   READ CURBO, P, MAIT: 50SUB 58   CLD 226   READ CURBO, P, MAIT: 50SUB 58   CLD 226   READ CURBO, P, MAIT: 50SUB 58   CLD 225   READ CURBO, P		CCH2		
228 FOR NE=1 TO 2:READ P_NAIT:GOSUB 30:N   279 READ LINES_CHORD_P_NAIT:GOSUB 30:N   270 READ LINES_CHORD_P_NAIT:GOSUB	210 READ LINE*, CHORD, P, WAIT: ? LINE*: 605U		746 IF PEEK (53279) = 3 THEN RUN	<mk></mk>
228 FOR NE=1 TO 2:READ P_NAIT:GOSUB 30:N   279 READ LINES_CHORD_P_NAIT:GOSUB 30:N   270 READ LINES_CHORD_P_NAIT:GOSUB	B 40	(SR)	750 IF PFFK (5.3279)=6 THEN 900	<iw></iw>
EXT ME:GOSUB 6:MAIT=10:GOSUB 50 223 FRAN CHORD, P, MAIT:10:GOSUB 50 225 FOR NEW 1 TO 2:READ P, WAIT:10:GOSUB 50:N EXT ME:GOSUB 6:MAIT-10:GOSUB		1007		
225 FRAD CHORD, P, WAIT: GOSUB 38:N EXT ME: GOSUB 64 MAIT-18: GOSUB 38:N EXT ME: GOSUB 64 MAIT-18: GOSUB 58:N CSD 86 ME: TO A: READ P, WAIT: GOSUB 58:N CSD 87 ME: TO A: READ P, WAIT: GOSUB 58:N CSD 88 ME: TO A: READ P, WAIT: GOSUB 58:N CSD 88 ME: TO A: READ P, WAIT: GOSUB 58:N CSD 88 ME: TO A: READ P, WAIT: GOSUB 58:N CSD 88 ME: TO A: READ P, WAIT: GOSUB 58:N CSD 88 ME: TO A: READ P, WAIT: GOSUB 58:N CSD 88 ME: TO A: READ P, WAIT: GOSUB 58:N CSD 88 ME: TO A: READ P, WAIT: GOSUB 58:N CSD 88 ME: TO A: READ P, WAIT: GOSUB 58:N CSD 88 ME: TO A: READ P, WAIT: GOSUB 58:N CSD 88 ME: TO A: READ P, WAIT: GOSUB 58:N CSD 88 ME: TO ME: TO A: READ P, WAIT: GOSUB 58:N CSD 88 ME:	ZZW FOR ME=1 TO Z:READ P,WAIT:GUSUB SW:N			SETZ
225 FRAD CHORD, P, WAIT: GOSUB 30:N EXT ME: GEREIR GORWALT-105:GOSUB 30:N EXT ME: GEREIR GORWALT-105:GOSUB 30:N EXT ME: GOSUB 40:0 206 FRAD LINES, CHORD, P, WAIT: FOSUB 30:N EXT ME: GOSUB 40:0 206 FRAD LINES, CHORD, P, WAIT: FOSUB 30:N EXT ME: GOSUB 40:0 206 FRAD LINES, CHORD, P, WAIT: FOSUB 30:N EXT ME: GOSUB 40:0 206 FRAD LINES, CHORD, P, WAIT: FOSUB 30:N EXT ME: GOSUB 40:0 206 FRAD LINES, CHORD, P, WAIT: FOSUB 30:N EXT ME: GOSUB 40:0 206 FRAD LINES, CHORD, P, WAIT: FOSUB 30:N EXT ME: GOSUB 40:0 207 FRAD LINES, CHORD, P, WAIT: FOSUB 30:N EXT ME: GOSUB 40:0 208 FRAD LINES, CHORD, P, WAIT: FOSUB 30:N EXT ME: GOSUB 40:0 209 FRAD LINES, CHORD, P, WAIT: FOSUB 30:N EXT ME: GOSUB 40:0 209 FRAD LINES, CHORD, P, WAIT: FOSUB 30:N EXT ME: GOSUB 40:0 209 FRAD LINES, CHORD, P, WAIT: FOSUB 30:N EXT ME: GOSUB 40:0 209 FRAD LINES, CHORD, P, WAIT: FOSUB 30:N EXT ME: GOSU	EXT ME:GOSUB 60:WAIT=10:GOSUB 50	(LE)	900 RUN "D: SOUND. 4": REM CASSETTEN BENUET	
225 FOR MEG. TO 2:READ P., WAIT:SCSUB 38:N CTS 2:38 FOR MEG. TO 1:READ P., WAIT:SCSUB 38:N CTS 2:38 FOR MEG. TO 2:READ P., WAIT:SCSUB 38:N CTS 2:38 FOR MEG. TO 2:READ P., WAIT:SCSUB 38:N CTS 2:38 FOR MEG. TO 2:READ P., WAIT:SCSUB 38:N CTS 2:38 FOR MEG. TO 2:READ P., WAIT:SCSUB 38:N CTS 2:38 FOR MEG. TO 2:READ P., WAIT:SCSUB 38:N CTS 2:38 FOR MEG. TO 2:READ P., WAIT:SCSUB 38:N CTS 2:38 FOR MEG. TO 2:READ P., WAIT:SCSUB 38:N CTS 2:READ CLINES, CHORD, P., WAIT:SCSUB 38:N CTS 3:READ CLINES, CHORD, P., WAIT:SCSUB 38:N CTS 3:REA	223 DEAD CHOON D WATT-COOLD AN	/75\	7FR MUSSEN HISR GRAPHICS M: END FINGEBEN	
EXT ME:GOSUB 69 WAIT-19:GOSUB 59 (VP) 269 FOR NE-1 TO 2:READ P,WAIT-59:GOSUB 50 (VP) 279 FOR BESTION S, WAIT-59:GOSUB 50 (VP) 280 FOR NE-1 TO 2:READ P,WAIT-59:GOSUB 50 (VP) 280 FOR NE-1 TO 3:READ P,WAIT-59:GOSUB 30 (VP) 280 FOR ME-1 TO 3:READ P,WAIT-59:GOSUB 3		VALUE /		CENT
1239 READ LINES, CHORD, P, MAIT; 67 **LINES:6* (YE)   1839 DATA, 49, 18, 30, 21, 30, 21, 50, 42, 20, 30, 25, 20, 20, 30, 20, 30, 20, 30, 20, 30, 20, 30, 20, 30, 20, 30, 30, 30, 30, 30, 30, 30, 30, 30, 3	225 FOR ME 1 TO 2: READ P, WAIT: GUSUB 30:N			COXY
1239 READ LINES, CHORD, P, MAIT; 67 **LINES:6* (YE)   1839 DATA, 49, 18, 30, 21, 30, 21, 50, 42, 20, 30, 25, 20, 20, 30, 20, 30, 20, 30, 20, 30, 20, 30, 20, 30, 20, 30, 30, 30, 30, 30, 30, 30, 30, 30, 3	EXT ME: GOSUB 60: WAIT-10: GOSUB 50	<lt></lt>	1000 DATA_Haens-chen_kleinging_al le	
USBS 48		,		< XE >
286 FOR MESI TO 5-READ P, WAIT-GOSUB 30-N			_	COM.
EXT ME:GOSUB 60: WAIT=C0:GOSUB 50: (NF) 250 FROM LINEs_CHORD, P. WAIT:F:SOSUB 30: (NF) 250 FROM ME=1 TO 2:READ P. WAIT:GOSUB 30: (NF) 251 FROM CHILD TO 2:READ P. WAIT:GOSUB 30: (NF) 252 FROM CHILD TO 2:READ P. WAIT:GOSUB 30: (NF) 253 FROM CHILD TO 2:READ P. WAIT:GOSUB 30: (NF) 254 FROM CHILD TO 2:READ P. WAIT:GOSUB 30: (NF) 255 FROM ME=1 TO 2:READ P. WAIT:GOSUB 30: (NF) 256 FROM ME=1 TO 2:READ P. WAIT:GOSUB 30: (NF) 257 FROM CLINEs, CHORD, P. WAIT:F? :P. LINES:G 058 40 258 FROM CLINES, CHORD, P. WAIT:F? :P. LINES:G 058 40 259 FROM ME=1 TO 4:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 4:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 4:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 4:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 2:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 2:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 2:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 2:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 2:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 2:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 2:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 2:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 2:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1	USUB 40	<ab></ab>		
EXT ME:GOSUB 60: WAIT=C0:GOSUB 50: (NF) 250 FROM LINEs_CHORD, P. WAIT:F:SOSUB 30: (NF) 250 FROM ME=1 TO 2:READ P. WAIT:GOSUB 30: (NF) 251 FROM CHILD TO 2:READ P. WAIT:GOSUB 30: (NF) 252 FROM CHILD TO 2:READ P. WAIT:GOSUB 30: (NF) 253 FROM CHILD TO 2:READ P. WAIT:GOSUB 30: (NF) 254 FROM CHILD TO 2:READ P. WAIT:GOSUB 30: (NF) 255 FROM ME=1 TO 2:READ P. WAIT:GOSUB 30: (NF) 256 FROM ME=1 TO 2:READ P. WAIT:GOSUB 30: (NF) 257 FROM CLINEs, CHORD, P. WAIT:F? :P. LINES:G 058 40 258 FROM CLINES, CHORD, P. WAIT:F? :P. LINES:G 058 40 259 FROM ME=1 TO 4:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 4:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 4:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 4:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 2:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 2:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 2:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 2:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 2:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 2:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 2:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 2:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 2:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1 TO 3:READ P. WAIT:GOSUB 30: (NF) 250 FROM ME=1	740 FOR MEST TO ASREAD P MATT-ROSID YOUR		23.30.23.60	<3C>
1939 READ LINES, CHORD, P, MAIT: 60SUB 38 N   1939 READ CHORD, P, MAIT: 60SUB 40   1040 DATA, 49, 35, 30, 35, 30, 32, 30, 31, 30, 32, 33, 30, 30, 32, 33, 30, 30, 33, 33, 30, 30, 33, 33, 30, 30		com.		
DSUB 40 2-86 FOR NET TO 2:READ P, WAIT: GOSUB 30:N EXT ME: GOSUB 6:M-WAIT: GOSUB 50 2-87 READ LINES, CHURD, P, WAIT: GOSUB 30:N EXT ME: GOSUB 6:M-WAIT: GOSUB 50:N EXT ME: GOSUB 6:M-WAIT: GOSUB 6:M-MAIT: GOSUB 50:N EXT ME: GOSUB 6:M-WAIT: GOSUB 6:M-MAIT:		(552		71785
DSUB 40 260 FOR RE=1 TO 2:READ P, WAIT:GOSUB 30:N EXT ME:GOSUB 60:WAIT=10:GOSUB 50 263 FOR RE=1 TO 2:READ P, WAIT:GOSUB 30:N EXT ME:GOSUB 60:WAIT=10:GOSUB 50:N EXT ME:GOSUB 60:WAIT=50:GOSUB 50:N EXT ME:GOSUB 60:WAIT=60:GOSUB 30:N EXT ME:GOSUB 60:WAIT=60:GOSUB 40:WAIT=60:GOSUB 50:FOR DECAY=10:WAIT=60:GOSUB 50:FOR DECAY=10:WAIT=60:GOSUB 50:FOR DECAY=10:WAIT=60:GOSUB 5	250 READ LINE*,CHORD,P,WAIT:? :? LINE*:6		1030 DATA_49,25,30,23,30,21,30,20,30,18,	
1046 DATA_SOUND_SILE_PORTdata_part_forms 50   10   10   10   10   10   10   10		(VE)	30.18.30.18.60	<hd></hd>
EXT MESGISUR 60 MAINTIGOSUR 50		1117		
285 FEAD CHREN, P, WAIT: GOSUB 50 EXT ME: GOSUB 68: WAIT=10: GOSUB 50 EXT ME: GOSUB 68: WAIT=10: GOSUB 50 CSD 40 280 FOR RE=1 TO 4: READ P, WAIT: GOSUB 30: N EXT ME: GOSUB 60: WAIT=30: GOSUB 50 CSD 70 READ LINEs, CHORD, P, WAIT: 7: C LINES: 6 CSUB 40 CSD FOR RE=1 TO 4: READ P, WAIT: GOSUB 30: N EXT ME: GOSUB 60: WAIT=30: GOSUB 50 CSD FOR RE=1 TO 6: READ P, WAIT: FOLIOSUB 30: N EXT ME: GOSUB 60: WAIT=30: GOSUB 50 CSD FOR RE=1 TO 6: READ P, WAIT: FOLIOSUB 30: N EXT ME: GOSUB 60: WAIT=30: GOSUB 50 CSD FOR RE=1 TO 6: READ P, WAIT: FOLIOSUB 30: N EXT ME: GOSUB 60: WAIT=30: GOSUB 40 CSD FOR RE=1 TO 2: READ P, WAIT: GOSUB 30: N EXT ME: GOSUB 60: WAIT=30: GOSUB 40 CSD FOR RE=1 TO 2: READ P, WAIT: GOSUB 30: N EXT ME: GOSUB 60: WAIT=30: GOSUB 40 CSD FOR RE=1 TO 2: READ P, WAIT: GOSUB 40 CSD FOR RE=1 TO 2: READ P, WAIT: GOSUB 40 CSD FOR RE=1 TO 2: READ P, WAIT: GOSUB 40 CSD FOR RE=1 TO 2: READ P, WAIT: GOSUB 40 CSD FOR RE=1 TO 2: READ P, WAIT: GOSUB 40 CSD FOR RE=1 TO 2: READ P, WAIT: GOSUB 40 CSD FOR RE=1 TO 2: READ P, WAIT: GOSUB 40 CSD FOR RE=1 TO 2: READ P, WAIT: GOSUB 40 CSD FOR RE=1 TO 2: READ P, WAIT: GOSUB 40 CSD FOR RE=1 TO 2: READ P, WAIT: GOSUB 40 CSD FOR RE=1 TO 2: READ P, WAIT: GOSUB 40 CSD FOR RE=1 TO 2: READ P, WAIT: GOSUB 40 CSD FOR RE=1 TO 8: READ CHORD, P, WAIT: GOSUB 40 CSD FOR RE=1 TO 8: READ CHORD, P, WAIT: GOSUB 40 CSD FOR RE=1 TO 8: READ CHORD, P, WAIT: GOSUB 40 CSD FOR RE=1 TO 8: READ CHORD, P, WAIT: GOSUB 40 CSD FOR RE=1 TO 8: READ CHORD, P, WAIT: GOSUB 40 CSD FOR RE=1 TO 8: READ CHORD, P, WAIT: GOSUB 40 CSD FOR RE=1 TO 8: READ CHORD, P, WAIT: GOSUB 40 CSD FOR RE=1 TO 8: READ CHORD, P, WAIT: GOSUB 40 CSD FOR RE=1 TO 8: READ CHORD, P, WAIT: GOSUB 40 CSD FOR RE=1 TO 8: READ CHORD, P, WAIT: GOSUB 40 CSD FOR RE=1 TO 8: READ CHORD, P, WAIT: GOSUB 40 CSD FOR RE=1 TO 8: READ CHORD, P, WAIT: GOSUB 40 CSD FOR RE=1 TO 8: READ CHORD, P, WAIT: GOSUB 40 CSD FOR RE=1 TO 8: READ CHORD, P, WAIT: GO				
285 FEAD CHURD, P, WAIT: GOSUB 38 (AND AND AND AND AND AND AND AND AND AND	EXT ME:GOSUB 60:WAIT=10:GOSUB 50	KLHD	oll	<tp></tp>
25 FOR ME=1 TO 2:READ P, MAIT: 60SUB 30:N EXT MESOSUB 60: MAIT=10:60SUB 50:N CNB 40 200 FOR ME=1 TO 4:READ P, MAIT: 60SUB 30:N EXT MESOSUB 60: MAIT=70: GOSUB 50:N EXT MESOSUB 60: MAIT=70: FILES: G SUB 40 300 FOR ME=1 TO 6: READ P, MAIT: 60SUB 30:N EXT MESOSUB 60: MAIT=70: FILES: G SUB 40 300 FOR ME=1 TO 8: READ P, MAIT: 60SUB 30:N EXT ME 320 READ LINES; CHORD, P, MAIT: 2: 7: LINES: G SUB 40 300 FOR ME=1 TO 2: READ P, MAIT: 60SUB 30:N EXT ME 320 READ LINES; CHORD, P, MAIT: 60SUB 30:N EXT ME 328 READ CHORD, P, MAIT: 60SUB 30:N EXT ME 339 READ P, MAIT: 60SUB 30:N EXT ME 340 READ P, MAIT: 60SUB 30:N EXT ME 340 READ D, MAIT: 60SUB 40:N EXT ME 340 READ D, MAIT: 60SUB 30:N EXT ME 340 READ D, MAIT: 60SUB 30:N EXT ME 340 READ D, MAIT: 60SUB 30:N EXT ME 340 READ D, MAIT: 60SUB 40:N EXT ME 340 READ D, MAIT: 60SUB 40:N EX			1858 DATA 49 30 30 33 33 33 40 42 32 30	
EXT ME: DOSUB 40: MATT=10: 50SUB 50: MS 279 READ LINES, CHORD, P. MAIT: 77 LINES: 6  OSUB 40  289 FOR ME=1 TO 4: READ P. MAIT: 50SUB 30: N EXT ME: GOSUB 60: MAIT=50SUB 50: N EXT ME: GOSUB 60: MAIT=50SUB 50: N EXT ME: GOSUB 40  300 FOR ME=1 TO 6: READ P. MAIT: 50SUB 30: N EXT ME S20 READ LINES, CHORD, P. MAIT: 7: 7: LINES: 6  OSUB 40  320 FOR ME=1 TO 6: READ P. MAIT: 50SUB 30: N EXT ME S30 FOR ME=1 TO 6: READ P. MAIT: 50SUB 30: N EXT ME S30 FOR ME=1 TO 6: READ P. MAIT: 50SUB 30: N EXT ME S30 FOR ME=1 TO 2: READ P. MAIT: 50SUB 30: N EXT ME S30 FOR ME=1 TO 3: READ P. MAIT: 50SUB 30: N EXT ME S30 FOR ME=1 TO 3: READ P. MAIT: 50SUB 30: N EXT ME S30 FOR ME=1 TO 3: READ P. MAIT: 50SUB 30: N EXT ME S30 FOR ME=1 TO 0: READ P. MAIT: 50SUB 30: N EXT ME S30 FOR ME=1 TO 0: READ P. MAIT: 50SUB 30: N EXT ME S30 FOR ME=1 TO 0: READ P. MAIT: 50SUB 30: N EXT ME S30 FOR ME=1 TO 0: READ P. MAIT: 50SUB 30: N EXT ME S30 FOR ME=1 TO 0: READ P. MAIT: 50SUB 30: N EXT ME S30 FOR ME=1 TO 0: READ P. MAIT: 50SUB 30: N EXT ME S30 FOR ME=1 TO 0: READ P. MAIT: 50SUB 30: N E		7545		/745
298 READ LINES, CHORD, P, WAIT: 7 : 7 LINES: 6 298 FOR ME=1 TO 4:READ P, WAIT: 60SUB 30:N EXT ME: 60SUB 40:0 300 FOR ME=1 TO 4:READ P, WAIT: 60SUB 30:N EXT ME: 50SUB 40:0 300 FOR ME=1 TO 6:READ P, WAIT: 60SUB 30:N EXT ME: 50SUB 40:0 300 FOR ME=1 TO 2:READ P, WAIT: 60SUB 30:N EXT ME: 350 READ LINES, CHORD, P, WAIT: 60SUB 30:N EXT ME: 350 READ LINES, CHORD, P, WAIT: 60SUB 30:N EXT ME: 350 READ LINES, CHORD, P, WAIT: 60SUB 30:N EXT ME: 350 READ LINES, CHORD, P, WAIT: 60SUB 30:N EXT ME: 350 READ LINES, CHORD, P, WAIT: 60SUB 30:N EXT ME: 350 READ LINES, CHORD, P, WAIT: 60SUB 30:N EXT ME: 350 READ LINES, CHORD, P, WAIT: 60SUB 30:N EXT ME: 350 READ LINES, CHORD, P, WAIT: 60SUB 30:N EXT ME: 350 READ LINES, CHORD, P, WAIT: 60SUB 40:0 350 READ CHORD, P, WAIT: 60SUB 50:FOR DECAY:15 TO 0 STEP -8.5:SUNDO V0; N(1), 19, DECAY:15 TO	245 FOR ME=1 TO 2:READ P,WAIT:60SUB 30:N		23*26*22*98	<2X>
279 READ LINES, CHORD, P, WAIT: 605UB 30:N EXT MEG SOSUB 60: MAIT: 7: LINES: 6 SUB 40: 300 FOR ME=1 TO 4:READ P, WAIT: 605UB 30:N EXT MEG SOSUB 60: MAIT: 7: 7: LINES: 6 SUB 40: 300 FOR ME=1 TO 6:READ P, WAIT: 605UB 30:N EXT ME 350 READ LINES, CHORD, P, WAIT: 605UB 30:N EXT ME 350 READ LINES, CHORD, P, WAIT: 605UB 30:N EXT ME 350 READ LINES, CHORD, P, WAIT: 605UB 30:N EXT ME 350 READ LINES, CHORD, P, WAIT: 605UB 30:N EXT ME 350 READ LINES, CHORD, P, WAIT: 605UB 30:N EXT ME 355 FOR ME=1 TO 2:READ P, WAIT: 605UB 30:N EXT ME 355 FOR ME=1 TO 2:READ P, WAIT: 605UB 30:N EXT ME 355 FOR ME=1 TO 2:READ P, WAIT: 605UB 30:N EXT ME 350 READ LINES, CHORD, P, WAIT: 605UB 30:N EXT ME 350 READ LINES, CHORD, P, WAIT: 605UB 40: (VP) 358 READ LINES, CHORD, P, WAIT: 605UB 40: (VP) 359 READ P, WAIT: 605UB 40: (VI) 360 READ CHORD, P, WAIT: 605UB 40: (VI) 379 READ P, WAIT: 605UB 40: (VI) 380 READ P, WAIT: 605UB 30: (VI) 380 READ CHORD, P, WAIT: 605UB 40: (VI) 380 READ CHORD, P, WAIT: 605UB 50: FOR DECAY: (VI) 380 READ CHORD, P, WAIT: 605UB 50: FOR DECAY: (VI) 380 READ CHORD, P, WAIT: 605UB 50: FOR DECAY: (VI) 380 READ CHORD, P, WAIT: 605UB 50: FOR DECAY: (VI) 380 READ CHORD, P, WAIT: 605UB 50: FOR DECAY: (VI) 380 READ CHORD, P, WAIT: 605UB 50: FOR DECAY: (VI) 380 READ CHORD, P, WAIT: 605UB 60: WAI	EXT ME: GOSUB AN: WATT=10: GOSUB 50	<mr></mr>	1060 DATALal-les_wohl_und_voll	<gw></gw>
128   A0		(10)		
1000 DATALREICHT_BISCUSUB_30: N	Z/W KEAD LINE* CHURD, F. WAITTY TY LINE*:8			
EXT ME:GOSUB 60:WAIT=30:GOSUB 50: (RS) OSUB 40 300 FOR ME=1 TO 6:READ P,WAIT:F0SUB 30:N EXT ME 320 READ LINE;F,CHORD,P,WAIT:F1: LINE;65 OSUB 40 330 FOR ME=1 TO 6:READ P,WAIT:F3: LINE;65 OSUB 40 330 FOR ME=1 TO 6:READ P,WAIT:F3: LINE;65 OSUB 40 330 FOR ME=1 TO 8:READ P,WAIT:F3: LINE;65 OSUB 40 330 FOR ME=1 TO 8:READ P,WAIT:F3: LINE;65 OSUB 40 340 FOR ME=1 TO 2:READ P,WAIT:F3: LINE;65 OSUB 40 34	OSUB 40	<yj></yj>	120	<kc></kc>
EXT ME:GOSUB 60:WAIT=30:GOSUB 50: (RS) OSUB 40 300 FOR ME=1 TO 6:READ P,WAIT:F0SUB 30:N EXT ME 320 READ LINE;F,CHORD,P,WAIT:F1: LINE;65 OSUB 40 330 FOR ME=1 TO 6:READ P,WAIT:F3: LINE;65 OSUB 40 330 FOR ME=1 TO 6:READ P,WAIT:F3: LINE;65 OSUB 40 330 FOR ME=1 TO 8:READ P,WAIT:F3: LINE;65 OSUB 40 330 FOR ME=1 TO 8:READ P,WAIT:F3: LINE;65 OSUB 40 340 FOR ME=1 TO 2:READ P,WAIT:F3: LINE;65 OSUB 40 34	200 FOR ME-1 TO ALDEAD P MAIT-COCUR TOLD		1889 DATA Reight dec. acht Rit SDIMD nich	
290 READ LINE*, CHORD, P, WAIT: 2: LINE*:6	The state of the s			4000
SUB 40   SERAP   SER	EXT ME:GOSUB 60:WAIT=30:GOSUB 50	<r\$></r\$>		< (B)
SUB 40   SERAP   SER	290 READ LINES.CHORD.P.WAIT: 2 : 2 LINES: 6		1090 DATA 42.35.30.35.30.35.30.35.30.35.	
100 BATA_masch=enamir_sech=zehn_Bit_dair		21/015		ZEMS
EXT ME		ZIMA		XEB2
S20 READ LINEs,CHORD,P,WAIT:7:7: LINEs:6   SUB 40   S38 FOR ME:1 TO B:READ P,WAIT:60SUB 38:N EXT ME	300 FOR ME=1 TO 6:READ P, WAIT: GOSUB 30:N		1100 DATA_mach-en_wir_sech-zehn_Bit_da r	
S20 READ LINEs,CHORD,P,WAIT:7:7: LINEs:6   SUB 40   S38 FOR ME:1 TO B:READ P,WAIT:60SUB 38:N EXT ME	EXT ME	CUES		<qe></qe>
15,33,15,30,38,668		1 41 7		
1128 DATA_AMIT_SIND_AFTER_ADP, WAIT: COSUB 30:N	320 READ LINES, CHURD, P, WAIT: 7:7 EINES: 6		1116 DHIHY4A-192-19-19-19-19-19-19-19-1	
SSO READ LINE\$, CHORD, P, MAIT: 2: ? LINE\$;6   CSUB 48	0SUB 40	<ya></ya>	15,33,15,32,15,30,30,0,60	<fm></fm>
SSO READ LINE\$, CHORD, P, MAIT: 2: ? LINE\$;6   CSUB 48				
SSO READ LINE*, CHORD, P, MAIT: 2 : 7 LINE*: 6		a a mind to		44.000
OSUB 400	EXT ME	<mi></mi>	**	< MITS
OSUB 400	350 READ LINE\$_CHORD_P.WAIT:? :? LINE\$:6		1130 DATA_49.30.30.33.30.33.60.42.32.30.	
148 DATA_Vielen_Dank_dem_PO-KEY_Chip   278		ZVCS		27TN
STA ME		1107		7217
363 READ CHORD,P,WAIT:GOSUB 40  365 FOR ME_1 TO 2:READ P,WAIT:GOSUB 30:N EXT ME  380 READ LINE\$,CHORD,P,WAIT:7:? LINE\$:6  OSUB 40  390 READ P,WAIT:GOSUB 30  400 READ CHORD,P,WAIT:GOSUB 40  400 READ CHORD,P,WAIT:GOSUB 40  400 READ P,WAIT:GOSUB 30  400 READ P,WAIT:GOSUB 40  400 READ CHORD,P,WAIT:GOSUB 40  400 READ CHORD,P,WAIT:GOSUB 40  400 READ CHORD,P,WAIT:GOSUB 40  400 READ P,WAIT:GOSUB 40  400 READ P,WA	360 FOR ME=1 TO 2:READ P, WAIT: GOSLO 30:N		1140 DATA_Vielen_Dank_dem_PO-KEY_Chip	
363 READ CHORD,P,WAIT:GOSUB 40  365 FOR ME_1 TO 2:READ P,WAIT:GOSUB 30:N EXT ME  380 READ LINE\$,CHORD,P,WAIT:7:? LINE\$:6  OSUB 40  390 READ P,WAIT:GOSUB 30  400 READ CHORD,P,WAIT:GOSUB 40  400 READ CHORD,P,WAIT:GOSUB 40  400 READ P,WAIT:GOSUB 30  400 READ P,WAIT:GOSUB 40  400 READ CHORD,P,WAIT:GOSUB 40  400 READ CHORD,P,WAIT:GOSUB 40  400 READ CHORD,P,WAIT:GOSUB 40  400 READ P,WAIT:GOSUB 40  400 READ P,WA	EXT ME	(TB)	4.4	<bc></bc>
SAS FOR NE_1 TO 2:READ P_WAIT:GOSUB 30:N EXT ME   SAS FOR NE_1 TO 2:READ P_WAIT:GOSUB 30:N   SAS READ LINE\$, CHORD, P, WAIT: 7: ? LINE\$:G   OSUB 40   OSUB 40   OSUB 40   SASO READ P_WAIT:GOSUB 30   AND READ CHORD, P, WAIT:GOSUB 40   AND ATA.44, 28, 30, 52, 53, 50   ATA.46, 28, 30   ATA.46,			1150 DATA AD 25 15 24 15	ZONS
SAB READ LINE\$, CHORD, P, MAIT: 7: P. LINE\$: 6   SUB 48   49   379 READ P, WAIT: 60SUB 30   4mb   410   42	363 KEHD CHOKD FF WHITE EGGSUB 46	1207		
188 DATA_49,25,70	365 FOR ME-1 TO 2: READ P.WAIT: GOSUB 30: N		1160 DATA_44,28,30,32,30	(SP)
188 DATA_49,25,70	EVT ME	/TON	1170 DATA_42_26_30_30_30	<da></da>
1200 DATA_49,37,C   CHB   390 READ P,WAIT:GOSUB 30   CHY   1210 DATA_47,35,D   CFT   1210 DATA_47,35,D   CFT   1210 DATA_47,35,D   CFT   1210 DATA_47,35,D   CFT   CFL   CFL   CFT				
1208 DATA_49, 37, C   CHB   1208 DATA_49, 37, C   CHB   1208 DATA_49, 35, D   CFT   CF	SEND KEAD LINES, CHORD, P, WAIT: ? :? LINES: G			
1218 DATA_47,35,D   1229 DATA_47,35,D   1229 DATA_47,35,D   1229 DATA_47,35,D   1229 DATA_47,35,E   1229 DATA_44,32,F   1229 DATA_42,36,G   1229 DATA_43,37,25,C   1229 DATA_42,37,25,C   1229 DATA_43,37,25,C   1229 DATA_42,37,25,C   1229 DATA_42,39,9   1329 DATA_42,39,			1200 DATA_49,37,C	<hb></hb>
### ### ##############################				<ft></ft>
### ### ##############################				
### ### ##############################	400 READ CHORD, P, WAIT: GOSUB 40	<yu></yu>		(FE)
### ### ##############################	410 READ P. MATT: GOSLIR 30	<lx></lx>	1230 DATA-44,32.F	(EE)
430 READ P,WAIT:GOSUB 30 (MB) 440 READ CHORD,P,WAIT:GOSUB 40 (ZC) 450 GOSUB 60:WAIT=10:GOSUB 50:FOR DECAY= 15 TO 0 STEP -0.5:SURRD V0,N(1),10,DECAY :NEXT DECAY 460 GRAPHICS 18:? #6:? #6; "anadurakkor de" 510 FOR ME=1 TO B:READ CHORD,P,LINE\$:POS ITION ME*2,10-ME:? #6;LINE\$:GOSUB 70:NEX T ME 530 FOR ME=8 TO 1 STEP -1:READ CHORD,P,L INE\$:POSITION ME*2,10-ME:? #6;LINE\$:GOSUB 70:NEX T ME 540 WAIT=15:GOSUB 50:POSITION 3,11:? #6; "BRUECKE" 550 FOR DECAY=15 TO 0 STEP -0.5:SUND V0 ,N(6),10,DECAY:NEXT DECAY 560 WAIT=15:GOSUB 50:POSITION 11,11:? #6; "START" 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO	·			
440 READ CHORD,P,WAIT:60SUB 40 450 GOSUB 60:WAIT=10:GOSUB 50:FOR DECAY= 15 TO 0 STEP -0.5:SURND V0,N(1),10,DECAY 15 TO 0 STEP -0.5:SURND V0,N(1),10,DECAY 2002 460 GRAPHICS 18:? #6:? #6; "anadurakkor de" 510 FOR ME=1 TO B:READ CHORD,P,LINE\$:POS ITION ME*2,10-ME:? #6;LINE\$:SOSUB 70:NEX T HE 530 FOR ME=8 TO 1 STEP -1:READ CHORD,P,L INE\$:POSITION ME*2,10-ME:? #6;LINE\$:SOSU B 70:NEXT ME 540 WAIT=15:GOSUB 50:POSITION 3,11:? #6; "DRUECKE" 550 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(6),10,DECAY:NEXT DECAY 560 WAIT=15:GOSUB 50:POSITION 11,11:? #6; "START" 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 400 STETOLOR 0,PEEK(20),10:IF PEEK(53279 1260 DATA_38,26,B 1270 DATA_49,28,3				
440 READ CHORD,P,WAIT:GOSUB 40 450 GOSUB 60:WAIT=10:GOSUB 50:FOR DECAY= 15 TO 0 STEP -0.5:SDRIND V0,N(1),10,DECAY 200 DATA_37,25,C 27A 21S TO 0 STEP -0.5:SDRIND V0,N(1),10,DECAY 22S DATA_48,28,0 23C DATA_48,28,0 23C DATA_48,28,0 23C DATA_44,38,0 23C DATA_45,38,0 23C DATA_45,38,0 23C DATA_45,38,0 23C DATA_45,38,0 23C DATA_45,38,0 23C DATA_47,35,0 23C DATA_48,28,0 23C DATA_44,38,0 23C DATA_48,28,0 23C DATA_48,32,6 23C D	430 READ P, WAIT: GOSUB 30	(MB)		(PL)
1270 DATA_37,25,C   CEV     15 TO 0 STEP -0.5:SURND V0,N(1),10,DECAY     18 TO 0 STEP -0.5:SURND V0,N(2)     18 TO 0 STEP -0.5:SURND V0,		<zc></zc>	1260 DATA_38,26,B	<fc></fc>
15 TO 0 STEP -0.5:SURND V0,N(1),10,DECAY :NEXT DECAY 460 GRAPHICS 18:? #6:? #6; "anadurakkor de"  510 FOR ME=1 TO 8:READ CHORD,P,LINE\$:POS ITION ME*2,10-ME:? #6;LINE\$:GOSUB 70:NEX T ME 530 FOR ME=8 TO 1 STEP -1:READ CHORD,P,L INE\$:POSITION ME*2,10-ME:? #6;LINE\$:GOSUB B 70:NEXT ME 540 WAIT=15:GOSUB 50:POSITION 3,11:? #6; "DRUECKE"  550 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 N(6),10,DECAY*NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 N(1),10,DECAY*NEXT DECAY 600 SETCOLOR 0,PEEK(20),18:IF PEEK(53279 1300 DATA_37,25,c (ZA) 1310 DATA_38,26,B (ZK) 1320 DATA_44,28,0 (XX) 1330 DATA_42,30,g (XX) 1340 DATA_44,32,f (ZH) 1350 DATA_44,32,f (ZH) 1350 DATA_44,32,d (SH) 1370 DATA_44,33,d (SH) 1370 DATA_44,32,d (SH) 1370 DATA_47,35,d (SH) 1370 DATA_49,28,0 (XX) 1380 DATA_42,28,0 (XX) 1380 DATA_48,28,0 (XX) 1380 DATA_49,28,0 (XX)	· ·	12.27		
:NEXT DECAY  460 GRAPHICS 18:? #6:? #6; "anadurakkor  de"  510 FOR ME=1 TO B:READ CHORD,P,LINE*:POS  ITION ME*2,10-ME:? #6;LINE*:GOSUB 70:NEX  T ME  530 FOR ME=8 TO 1 STEP -1:READ CHORD,P,L  1350 DATAL42,30,g  (XY)  1340 DATAL42,30,g  (XY)  1350 DATAL42,30,g  (XY)  1350 DATAL42,30,g  (XY)  1350 DATAL43,32,f  (ZU)  1350 DATAL45,33,e  (ZU)  1350 DATAL45,33,e  (ZU)  1350 DATAL45,33,e  (ZU)  1350 DATAL47,35,d  (BI)  1350 DATAL49,37,c  (BI)  1350	404 GUSUR GA: MAIT=IN: GUSUB 20: FUR DECAY=			
1310 BATA_38_26_b	15 TO 0 STEP -0.5: SOUND VO.N(1).10.DECAY		1.500 DATA_57,25,6	<4A>
460 GRAPHICS 18:? #6; **anadur*akkor*  de"		< 0.045	1310 DATA 38 26 - h	(ZN)
Side For ME=1 to B:READ CHORD,P,LINE\$:POS   1330 DATA_42,30,g   (XT)   Side For ME=1 to B:READ CHORD,P,LINE\$:POS   1340 DATA_44,32,f   (ZH)		COLOR		
Side For ME=1 TO B:READ CHORD,P,LINE\$:POS   1348 DATA_44,32,f   (7H)   1710 ME*2,10-ME:? #6;LINE\$:GOSUB 70:NEX   1358 DATA_44,32,f   (7H)   1358 DATA_45,33,e   (7H)   1358 DATA_45,33,e   (7H)   1358 DATA_45,32,f   (7H)   1358 DATA_45,3	460 GRAPHICS 18:? #6:? #6; "adur_akkor			< MMC>
Si8 FOR ME=1 TO 8:READ CHORD,P,LINE\$:POS   ITION ME*2,10-ME:? #6;LINE\$:GOSUB 70:NEX   T ME		(JK)	1330 DATA_42,30.q	<xt></xt>
ITION ME*2,10-ME:? #6;LINE*:GOSUB 70:NEX T ME  530 FOR ME=8 TO 1 STEP -1:READ CHORD,P,L INE*:POSITION ME*2,10-ME:? #6;LINE*:GOSU B 70:NEXT ME  540 WAIT=15:GOSUB 50:POSITION 3,11:? #6; "DRUECKE"  550 FOR DECAY=15 TO 0 STEP -8.5:SOUND V0 ,N(6),10,DECAY:NEXT DECAY  570 FOR DECAY=15 TO 0 STEP -8.5:SOUND V0 ,N(1),10,DECAY=15 TO 0 STEP -8.5:SOUND V0 ,N(1),10,DECAY=NEXT DECAY  570 FOR DECAY=15 TO 0 STEP -8.5:SOUND V0 ,N(1),10,DECAY=NEXT DECAY  570 FOR DECAY=15 TO 0 STEP -8.5:SOUND V0 ,N(1),10,DECAY=NEXT DECAY  570 FOR DECAY=15 TO 0 STEP -8.5:SOUND V0 ,N(1),10,DECAY=NEXT DECAY  570 FOR DECAY=15 TO 0 STEP -8.5:SOUND V0 ,N(1),10,DECAY=NEXT DECAY  570 FOR DECAY=15 TO 0 STEP -8.5:SOUND V0 ,N(1),10,DECAY=NEXT DECAY  570 FOR DECAY=15 TO 0 STEP -8.5:SOUND V0 ,N(1),10,DECAY=NEXT DECAY  570 SETCOLOR 0,PEEK(20),18:IF PEEK(53279 ]  1350 DATA_45,33,e  1350 DATA_47,35,d  (B1)  1370 DATA_49,37,c  20010 ? .*(CTL R)(CTL R)(		130132		
T ME 530 FOR ME=8 TO 1 STEP -1:READ CHORD,P,L INE\$:POSITION ME*2,10-ME:? #6;LINE\$:SOSU B 70:NEXT ME 540 WAIT=15:GOSUB 50:POSITION 3,11:? #6; "DRUECKE" 550 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(6),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 SETCOLOR 0,PEEK(20),18:IF PEEK(53279 ) 1360 DATA_47,35,d 1370 DATA_49,37,c 20010 ?,"(CTL R)(CTL R			· ·	
T ME 530 FOR ME=8 TO 1 STEP -1:READ CHORD,P,L INE\$:POSITION ME*2,10-ME:? #6;LINE\$:SOSU B 70:NEXT ME 540 WAIT=15:GOSUB 50:POSITION 3,11:? #6; "DRUECKE" 550 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(6),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 SETCOLOR 0,PEEK(20),18:IF PEEK(53279 ) 1360 DATA_47,35,d 1370 DATA_49,37,c 20010 ?,"(CTL R)(CTL R	ITION ME+2.10-ME:? #6:LINES:GOSUB 70:NEX		135@ DATA_45,33,e	<zu></zu>
530 FOR ME=8 TO 1 STEP -1:READ CHORD,P,L INE#:POSITION ME*2,10-ME:? #6;LINE#:GOSU B 70:NEXT ME 540 WAIT=15:GOSUB 50:POSITION 3,11:? #6; "DRUECKE" 550 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(6),10,DECAY:NEXT DECAY 560 WAIT=15:GOSUB 50:POSITION 11,11:? #6 ;"START" 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 SETCOLOR 0,PEEK(20),18:IF PEEK(53279 ) 1370 DATA_49,37,C 20000 POSITION 2,20:POKE 201,4:POKE 752, 1 20010 7,"(CTL R)(CTL R)(CT		CNRIN	1360 DATA_47.35_d	<bi></bi>
INE#:POSITION ME#2,10-ME:? #6;LINE#:GOSU B 70:NEXT ME 540 WAIT=15:GOSUB 50:POSITION 3,11:? #6; "DRUECKE" 550 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(6),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ;"START" 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 570 SETCOLOR 0,PEEK(20),18:IF PEEK(53279 1 120000 POSITION 2,20:POKE 201,4:POKE 752,  120010 7,"(CTL 0)(CTL N)(CTL			·	
INE#:POSITION ME*2,10-ME:? #6;LINE#:SOSU B 70:NEXT ME 540 WAIT=15:GOSUB 50:POSITION 3,11:? #6; "DRUECKE" 550 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(6),10,DECAY:NEXT DECAY 550 WAIT=15:GOSUB 50:POSITION 11,11:? #6 ;"START" 570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 600 SETCOLOR 0,PEEK(20),18:IF PEEK(53279 ] <a "start"="" #6="" ,n(1),10,decay:next="" ,n(6),10,decay:next="" -0.5:sound="" 0="" 0,peek(20),10:if="" 11,11:?="" 50:position="" 550="" 570="" 600="" ;="" ]<="" decay="" druecke"="" for="" href="https://documents.com/bit/10-mil/star-1&lt;/td&gt;&lt;td&gt;530 FOR ME=8 TO 1 STEP -1: READ CHORD.P.L&lt;/td&gt;&lt;td&gt;&lt;/td&gt;&lt;td&gt;&lt;/td&gt;&lt;td&gt;CM&gt;&lt;/td&gt;&lt;/tr&gt;&lt;tr&gt;&lt;td&gt;## ## ## ## ## ## ## ## ## ## ## ## ##&lt;/td&gt;&lt;td&gt;&lt;/td&gt;&lt;td&gt;&lt;/td&gt;&lt;td&gt;20000 POSITION 2.20: POKE 201.4: POKE 752.&lt;/td&gt;&lt;td&gt;&lt;/td&gt;&lt;/tr&gt;&lt;tr&gt;&lt;td&gt;540 WAIT=15:GOSUB 50:POSITION 3,11:? #6; " peek(53279="" pre="" setcolor="" step="" to="" v0="" wait="15:GOSUB">  20010 ? ,"(CTL 0)(CTL R)(CTL R)(CTL</a>		10115		(PR)
"DRUECKE"  550 FOR DECAY=15 TO 0 STEP -8.5:SDUND V0 ,N(6),10,DECAY:NEXT DECAY  560 WAIT=15:GOSUB 50:POSITION 11,11:? #6 ; "START"  570 FOR DECAY=15 TO 0 STEP -8.5:SDUND V0 ,N(1),10,DECAY:NEXT DECAY  570 FOR DECAY=15 TO 0 STEP -8.5:SDUND V0 ,N(1),10,DECAY:NEXT DECAY  600 SETCOLOR 0,PEEK(20),10:IF PEEK(53279 ] 1 R)(CTL R	B /O:NEXT ME	<gh></gh>		21.172
"DRUECKE"  550 FOR DECAY=15 TO 0 STEP -8.5:SDUND V0 ,N(6),10,DECAY:NEXT DECAY  560 WAIT=15:GOSUB 50:POSITION 11,11:? #6 ; "START"  570 FOR DECAY=15 TO 0 STEP -8.5:SDUND V0 ,N(1),10,DECAY:NEXT DECAY  570 FOR DECAY=15 TO 0 STEP -8.5:SDUND V0 ,N(1),10,DECAY:NEXT DECAY  600 SETCOLOR 0,PEEK(20),10:IF PEEK(53279 ] 1 R)(CTL R	540 WAIT=15:GOSUB 50:POSITION 3.11:2 #4:		ZWW10 ? , T(CTL Q) (CTL R) (CTL R) (CTL R) (C	
550 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(6),10,DECAY:NEXT DECAY  560 WAIT=15:GOSUB 50:POSITION 11,11:? #6 ; "START"  570 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY  600 SETCOLOR 0,PEEK(20),18:IF PEEK(53279 ] <> 600 SETCOLOR 0,PEE		(7P)		
N(6),10,DECAY:NEXT DECAY		(LD)		
N(6),10,DECAY:NEXT DECAY	550 FOR DECAY=15 TO 0 STEP -0.5:SDUND V0		COLE RESCOIL RESCOIL RESCOIL RESCOIL	
560 WAIT=15:GOSUB 50:POSITION 11,11:? #6 ; "START"		(CT)	R) (CTL R) (CTL R) (CTL R) (CTL R) (CTL R) (CTL	
;"START"		(61)		
578 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 600 SETCOLOR 0,PEEK(20),18:IF PEEK(53279 1<>600 SETCOLOR 0,PEEK(20),18:IF PEEK(53279 1<>600 SETCOLOR 0,PEEK(20),18:IF PEEK(53279 1	360 WAIT=15:60SUB 50:POSITION 11,11:? #6			
578 FOR DECAY=15 TO 0 STEP -0.5:SOUND V0 ,N(1),10,DECAY:NEXT DECAY 600 SETCOLOR 0,PEEK(20),18:IF PEEK(53279 1<>600 SETCOLOR 0,PEEK(20),18:IF PEEK(53279 1<>600 SETCOLOR 0,PEEK(20),18:IF PEEK(53279 1	: "START"	<cd></cd>	CIL RY(CTL E)"	<at></at>
N(1),10,DECAY:NEXT DECAY			20020 ? -*(SHIFT a) Demacks START from	
600 SETCOLOR 0, PEEK (20), 18: IF PEEK (53279 (3J) 7L R3 (CTL R)				/P/15
600 SETCOLOR 0, PEEK (20), 18: IF PEEK (53279   20030 ? , "(CTL 2) (CTL R)		ZIMS	erter_tourt = }"	< PW >
1<>6 THEN 600 (JJ) TL RECTL RECTL RECTL RECTL RECTL RE		(OC)		
	,N(1),10,DECAY:NEXT DECAY	(OC)	20030 ? ,"(CTL Z)(CTL R)(CTL R)(CTL R)(C	
665 RESTORE 22000 (CC) Listing 2. Ein Lied zum Mitsingen. (Fortsetzung)	,N(1),10,DECAY:NEXT DECAY 600 SETCOLOR 0,PEEK(20),10:IF PEEK(53279			
	,N(1),10,DECAY:NEXT DECAY 600 SETCOLOR 0,PEEK(20),10:IF PEEK(53279	<jj></jj>	TL RESETTE RESCOTE RESCOTE RESCOTE RESCOTE RE	



(CTL R)(CTL R)(CTL R)(CTL R)(CTL R)(CTL R)(CTL R)(CTL R)(CTL R)(CTL R)(CTL R)(CTL L R)(CTL R)(CTL R)(CTL R)(CTL R)(CTL R)	T 21020 ? ,"(SHIFT =)_BEGLEITUNG_MITADUR_A	<ja></ja>
	(BM) 21030 ? ,"(CTL Z)(CTL R)(CTL R)(CTL R)(C	
	TL R) (CTL R) (CTL R) (CTL R) (CTL R) (CTL R)	
	<ts> (CTL R)(CTL R)(CTL R)(CTL R)(CTL R)(CTL</ts>	
21000 GRAPHICS 0:SETCOLOR 2,9,0:SETCOLOR	113 1010 113 1010 111 1011 111 1011	
4,9,8:SETCOLOR 1,9,12:POKE 752,1:POKE 8		
2,2:POKE 83,39:POKE 201,3		<hh></hh>
21010 POSITION 2,0:? ,"(CTL 0)(CTL R)(CT	T 21040 POKE 77,0:RETURN <	<ak></ak>
L RECOTL RECOTL RECOTL RECOTL RECOTL REC	Z2000 DATA_0,18,190,120,21000,21040,100,	
CTL R) {CTL R) {CTL R} {CTL R} {CTL R} {CTL R	R 210,30,74,210,220,40,54,1000,1010	<rz></rz>
CTL RECTL RECTL RECTL RECTL RECTL	L	
RECOTE RECOTE RECOTE RECOTE RECOTE RECO		
TL RECOTL RECOTL RECOTL RECOTL RECOTL EE	Listing 2. Ein Lied zum Mitsingen. (Schluß)	

B REM SOUND.4	1	:7	<du></du>
c) 1985 by Peter Gerstner 11/10/85	<ue></ue>	370 X=49:SOUND 8,N(X),10,14:SOUND 1,N(X-	
1 REM	<'UQ>	3),10,6:SOUND 2,N(X-7),10,6:SOUND 3,N(X-	
2 REM	<ur></ur>	12),10,6	<yt></yt>
3 REM	(US)	380 WAIT=60:605UB 50:005UB 60	<ie></ie>
10 DIM TYPE\$(40),N(50):V0=0:V1=1:V2=2:V3		390 LINE=340:GOSUB 22000	<ue></ue>
=3:POKE 82,2:2 "KESC CTL ->":60TO 100	<pd></pd>	480 TYPE\$ "OK, _zurueck_zum_normalen_SOUN	
50 POKE 540, WAIT	<vy></vy>	D":HOR=5:VER=6:GOSUB 70	<xt></xt>
52 IF PEEK (548) (>0 THEN 52	<00>	410 TYPE\$-"Befehlaundamachen_einenaVergl	
54 RETURN	<hl></hl>	eich. ": HOR=3: VER-8: GOSUB 70: GOSUB 50	<ut></ut>
60 FOR OFF-0 TO 3:SOUND OFF,0,0,0:NEXT B		420 TYPEs="Dieatrefste_Note_dieawiramit"	- 5145
FF: RETURN	<km></km>	:HOR=6:VER=10:GOSUB 70	<rm></rm>
70 LINE=LEN(TYPE\$):POSITION HOR, VER	<ng></ng>	430 TYPE\$="mit_DISTORTION_10_bekommen_ko	es mak
72 FOR ME=1 TO LINE: 7 TYPE#(ME, ME); : IF T		ennen, ": HDR-3: VER=12: GOSUB 70: GOSUB 50	<mm></mm>
YPE\$(ME, ME)=""" THEN 76	<dl.></dl.>	435 TYPE\$ "ist_ein_tiefes_'B'.":HOR=11:V	/L TS
74 REM SOUND 0,25,4,6:FOR DECAY=6 TO 0 S		ER=14: GOSUB 70: GOSUB 50	<lt></lt>
TEP -0.5:SOUND 0,10,0,DECAY:NEXT DECAY	<ag></ag>	440 TYPE\$="SOUND.V0,255,10,8":HOR=12:VER	
76 NEXT ME:RETURN	<nz></nz>	=16:GOSUB 70:GOSUB 60:SOUND V0,255,10,8:	<fv></fv>
100 DATA_14,15,16,17,18,19,21,22,23,24,2	*	GOSUB 50: GOSUB 60	<80>
6,27,29,31,33,35,37,40,42,45,47,50,53,57	comes 1	450 GUSUB 20000: GUSUB 21000	/ Abr/
,60,64,68,72,76,81,85,91,96	< MM>	500 TYPE*-"Vergleichen_wir_nun_zwei_Soun ds": MOR=4: VER=6: 60SUB 70: 60SUB 50	<xf></xf>
110 DATA_102,108,114,121,128,136,144,153	(50)	510 TYPE\$="\$0UND_V0,255.10,8":HOR=12:VER	/AL/
,162,173,182,193,204,217,230,243,255 120 GOSUB 21000:FOR X=1 TO 50:READ IT:NO	<fp></fp>	=8:605U8 70:605UB 50	<je></je>
X) = IT: NEXT X	<ay></ay>	520 SOUND V0.255,10,8:60SU8 50:60SUB 60:	Chi.
130 TYPE="Bitte_den_Begleittext_fuer_SO	VH47	60SUB 58	<ys></ys>
UND. 4.": HOR=3: VER=9: GOSUB 76	(CZ)	538 TYPE*="SOUND.V0,33,12,8":HOR=12:VER=	1107
140 TYPE\$="lesen_und_erst_die_START-Tast	1027	10: GOSUB 70: GOSUB 50	<me></me>
e_druecken":HOR=1:VER=11:GOSUB 70	<8X>	540 SOUND V0,33,12,8:60SUB 50:60SUB 60:6	VIII /
150 TYPEs="wenn_es_im_Text_verlangt_wird	'WAZ	OSUB 50	< 43>
.":HOR=5:VER=13:GOSUB 70	<ua></ua>	550 TYPEs="Die_Tonqualitaet_ist_verschie	
160 GOSUB 20000: GOSUB 21000	<xp></xp>	den_aber":HOR=2:VER=12:GOSUB 78	<ad></ad>
190 TYPE\$="DURA_AKKORDE:":HOR-14:VER=6:G		560 TYPE\$="die_aktuelle_PITCH_ist_das_ql	
DSUB 70:GOSUB 50:? :? :LIST 200.210	(HF)	eiche_R. ": HOR=2: VER=14: GOSU8 70: GOSUB 50	<uz></uz>
200 FOR X=13 TO 50: SOUND 0.N(X),10,14:SD		570 POSITION 12,8:? "SOUND.V0,255,10,8":	
UND 1,N(X-4),10,6:SOUND 2,N(X-7),10,6:SO		POSITION 12,10:? "SOUND_VE,33,12,8":SOUN	
UND 3,N(X-12),10,6	<x6></x6>	D V0,255,10,8	<kf></kf>
210 GOSUB 60:NEXT X	(PB)	580 GOSUB 50: GOSUB 40: GOSUB 50: POSITION	
215 IF LINE(>200 THEN ? :LIST 220,230	<fu></fu>	12,8:7 "SOUNDAV0,255,10,8"	<yd></yd>
220 FOR X=50 TO 13 STEP -1:SOUND 0,N(X),		590 POSITION 12,10:7 "SOUND_V0,33,12,8":	
10,14:SOUND 1,N(X-4),10,6:SOUND 2,N(X-7)		SOUND V2,33,12,8:60SUB 50:60SUB 60:60SUB	
,10,6:SOUND 3,N(X-12),10,6	<kh></kh>	50	<10>
230 GOSUB 60:NEXT X	<pf></pf>	600 POSITION 12,8:? "SOUND_V0,255,10,8":	
240 LINE=200:50SUB 22000	<rr></rr>	SOUND V0,255,10,8:POSITION 12,10:7 "SOUN	
250 TYPE\$="MOLL_AKKORDE: ": HOR=14: VER=6:G		D_V1,33,12,8":50UND V1,33,12,8	(NS)
OSUB 70:60SUB 50:7 :LIST 260,270	<1K>	610 GUSUB 50:GUSUB 60:LINE=570:GUSUB 220	er iller
260 FOR X=13 TO 50:SOUND 6,N(X),10,14:SO		DO TABLE HILL IN LAND AND THE PROTECTION OF	<vz></vz>
UND 1,N(X-3),10,6:SOUND 2,N(X-7),10,6:SO	44.5	620 TYPE\$="Wirakoennenimit_DISTORTION_12	4071
UND 3,N(X-12),10,6	<ut></ut>	":HOR=5:VER=6:GOSUB 70	<rj></rj>
270 GOSUB 60:NEXT X	<pn></pn>	630 TYPEs="anstatt_DISTORTION_10_noch_ti	CHES
280 IF LINE<>260 THEN ? :LIST 290,300	<pa></pa>	efere": HOR-3: VER=8: GOSUB 78: GOSUB 58	(NE)
290 FOR X=50 TO 13 STEP -1:SOUND 0,N(X),		635 TYPE\$="BASS_NOTEN_bekommen.": HOR=11:	/CD>
10,14:SOUND 1,N(X-3),10,6:SOUND 2,N(X-7)	(30)	VER=10:GOSUB 70:GOSUB 50	<sp></sp>
,10,6: SOUND 3,N(X-12),10,6	<10>	640 TVPE\$="/UM_BEISPIEL:":HOR-14:VER=12:	<yq></yq>
300 GOSUB 60:NEXT X 310 LINE 260:GOSUB 22000	<pa></pa>	GOSUB 70:GOSUB 50 700 DATA_25,27,28,30,31,33,36,37,40,42,4	1 1007
320 X=49	<ua></ua>	5,48,51,52,57,60,63,67,72,75,82,85,90,97	
330 TYPE\$="C_DUR":HOR=17:VER=6:60SUB /0:	(MV)	102	<fn></fn>
GOSUB 50:7 :LIST 340:7	(AB>	710 FOR BASS=1 TO 25:READ PO:FOR DECAY=1	41.445
340 X=49:SOUND 0,N(X),10,14:SOUND 1,N(X-	THE	5 TO 8 STEP -1	<at></at>
4),10,6:SOUND 2,N(X-7),10,6:SOUND 3,N(X-		720 POSITION 12,14:? "SOUNDAVO,"; PO; ",12	
12),10,6	<ac></ac>	,"; DECAY; "A": SOUND VØ, PØ, 12, DECAY: NEXT D	
350 WAIT=60:GOSUB 50:GOSUB 60	<hz></hz>	ECAY: NEXT BASS	<jh></jh>
360 IF LINE(>340 THEN TYPE#="C.MOLL":HOR		Listing 3. Verschiedene Akkorde. Bitte mit dem Nam	
=17:VER=12:60SUB 70:60SUB 50:7 :LIST 370		*D:SOUND.4« auf Diskette speichern.	1611
	,	-D.OOOHD.7- dui Diskette Speichern.	

	750 RESTORE 700:LINE=710:GOSUB 22000	<cb>  </cb>	TE RECOTE RECOTE RECOTE RECOTE RECOTE RE	
	800 TYPEs="Das_naechste_Programm_erlaubt		(CTL R) (CTL R) (CTL R) (CTL R) (CTL R)	
ı	Auns": HOR=4: VER=6: GOSUB 78	<xp></xp>	RECOTE RECOTE RECOTE EET	<jm></jm>
ı	810 TYPE\$="verschiedene_Sounds_anzuhoere	1	21020 ? ,"{SHIFT =}_Der_SOUND_Befehl_{SH	
1	n, wenn": HOR=3: VER=8: GOSUB 70	<dk></dk>		V
ł	820 TYPE\$="wir_mit_den_Joystick_PITCH,_V		21030 ? ,"(CTL Z)(CTL R)(CTL R)(CTL R)(C	
ł	OLUME, ": HOR=3: VER=10: GOSUB 70	<x0></x0>	TE RECOTE RECOTE RECOTE RECOTE RECOTE RE	
ı	830 TYPE\$="und_DISTORTION_versendern_":H		(CTL R) (CTL R) (CTL R) (CTL R) (CTL R) (CTL	
ı	OR=7: VER=12: 60SUB 70: GOSUB 20000	<dg></dg>	R) (CTL R) (CTL R) (CTL R) (CTL C)"	<lh></lh>
ı	840 RUN "D: SOUND. 5": REM CASSETTEN BENUET	1	*	<ak></ak>
ı	ZER MUESSEN HIER GRAPHICS 0:END EINGEBEN	T.	22000 POSITION 2,19:POKE 201,5:POKE 752,	47 11 17
ı	ANSTELLE VON RUN "D:SOUND.5"	(RG)	1	< MM >
ı	20000 POSITION 2,20: POKE 201.4: POKE 752.	CII.C.	22010 ? ."(CTL Q) (CTL R) (CTL R) (CTL R) (C	
ı	1	<pr></pr>	TL R) (CTL R) (CTL R) (CTL R) (CTL R)	
ı	20010 ? "(CTL Q)(CTL R)(CTL R)(CTL R)(C		CCTL RECTL RECTL RECTL RECTL RECTL	
1	TL RY (CTL R) (CTL R) (CTL R) (CTL R)	1	R) (CTL R) (CTL R) (CTL R) (CTL R) (CTL	
J	(CTL R) (CTL R) (CTL R) (CTL R) (CTL	1		/ W 7 %
1	R) (CTL R) (CTL R) (CTL R) (CTL R) (CTL R)	1	L R) (CTL R) (CTL R) (CTL R) (CTL E) "	< X Z >
ı			22015 ? ,"(SHIFT =)_OPTION_=Wiederholu	*****
ı	L RYCTL RYCTL RYCTL RYCTL RYCTL RYCTL RYC		ngaa(SHIFT =>"	<uv></uv>
ı	CTL R) (CTL E) 4	<at></at>	22020 ? "(SHIFT =) START " Weiter	
ł	20020 ? ,"(SHIFT =)_Druecke_START_fuer_w		AAAA (SHIFT =)"	<je></je>
ı	eiter=(SHIFT =>"	<pw></pw>	22030 7 ,"(CTL Z)(CTL R)(CTL R)(CTL R)(C	
ı	20030 7 ,"(CTL Z)(CTL R)(CTL R)(CTL R)(C		TL R3 (CTL R) (CTL R) (CTL R) (CTL R)	
ı	TL R) (CTL R) (CTL R) (CTL R) (CTL R)		(CTL R) (CTL R) (CTL R) (CTL R) (CTL	
1	(CTL R) (CTL R) (CTL R) (CTL R) (CTL		R) (CTL R) (CTL R) (CTL R) (CTL R) (CTL R) (CTL	
ı	R) (CTL R) (CTL R) (CTL R) (CTL R) (CTL R) (CTL		L R) (CTL R) (CTL R) (CTL R) (CTL R) (CTL C)*	<zg></zg>
ı	L R) (CTL R) (CTL R) (CTL R) (CTL R) (		22040 IF PEEK (53279) -6 THEN POKE 755,2:6	
ı	CTL RECTL CE"	<bw></bw>	OSUB 21000: RETURN	<tv></tv>
ı	20048 IF PEEK (53279) <>6 THEN POKE 755,3:		22050 IF PEEK (53279) = 3 THEN POP : POKE 75	
ı	POKE 755,2:GOTO 20046	<>>>	5.2:POSITION 2.19:? "(ESC SHIFT DEL)(ESC	
ı	20050 POKE 755,2:RETURN	<cs></cs>	SHIFT DEL) (ESC SHIFT DEL) (ESC SHIFT DEL	
	21000 GRAPHICS 0: SETCOLOR 2.9.0: SETCOLOR		>"::GOTO LINE	<0M>
	4,9,0:SETCOLOR 1,9,12:POKE 752,1:POKE 8			<xl></xl>
	2,2:POKE 83,39:POKE 201,8	<in></in>		
	21818 ? ,"(CTL Q)(CTL R)(CTL R)(CTL R)(C		Listing 3. Verschiedene Akkorde. (Schluß)	

Ø REM SOUND.5		240 IF STRIG(0)=0 THEN DISTORTION=DISTOR	
(c) 1985 by Peter Gerstner 13/10/85	<@J>	TION+2:GOSUB DELAY	<ym></ym>
1 REM	<b><uq></uq></b>	250 IF DISTORTION=16 THEN DISTORTION=0	<pm></pm>
2 REM	<ur></ur>	260 IF STICK(0)=7 THEN VOLUME=VOLUME+1:I	
140 GRAPHICS 2:POKE 752,1:POKE 710,145:P		F VOLUME>15 THEN VOLUME=15	<bx></bx>
OKE 712,145:POKE 201,101? "(ESC CTL -)"	<nx></nx>	270 IF STICK(0)=11 THEN VOLUME=VOLUME-1:	
150 ? "JOYSTICK-auf/abassas=aTONHOEHE"	<12>	IF VOLUMEKØ THEN VOLUME=Ø	<fu></fu>
168 ? "JOYSTICKatriggerassas" VERZERRUN		280 IF PEEK (764) <> 255 THEN GRAPHICS 0:PG	
6*	<zr></zr>	KE 764.255:POKE 752.0:RUN "D:SOUND.6"	<fa></fa>
170 ? "JDYSTICK_links/rechts_=_LAUTSTAER		290 POSITION 2,417 #6; "sound.0,"; PITCH; "	
KE"	<wv></wv>	." : DISTORTION: "." : VOLUME: "AAAA"	<cm></cm>
180 ? "DRUECKELEINELBELIEBIGELTASTELFUER		300 SOUND 0.PITCH.DISTORTION, VOLUME	<2B>
▲END";	<vw></vw>	310 SETCOLOR 0, DISTORTION, 10: SETCOLOR 1.	
190 POKE 764,255	<wz></wz>	VOLUME, 10: POKE 77,0: GOTO 230	<sn></sn>
208 VOLUME=2	<r0></r0>	320 FOR WAIT=1 TO 100:NEXT WAIT:RETURN	<yj></yj>
210 DISTORTION=10	<kq></kq>	330 REM CASSETTEN BENUETZER MUESSEN IN Z	
220 DELAY=320	<aq></aq>	EILE 280 'RUN "D: SOUND. 6" ' AENDERN IN '	
230 IF STICK(0)=14 THEN PITCH=PITCH+1: IF		GRAPHICS 0: END '	<0A>
PITCH>255 THEN PITCH=255	<xk></xk>		
235 IF STICK(0)=13 THEN PITCH=PITCH-1: IF	4	Listing 4. Experimentieren mit dem SOUND-Befehl.	Bitte
PITCH(0 THEN PITCH=0	<dp></dp>	mit dem Namen »D:SOUND.5« auf Diskette speicher	

		<u> </u>	
Ø REM SOUND.6		191 REM	<bd></bd>
c) 1985 by Peter Gerstner 03/86/85	<ln></ln>	200 FOR P0=10 TO 2 STEP -0.02:VOL-P0/2:S	
1 REM	<uq></uq>	OUND V8.P8.8.VOL:SOUND V1.P8+1.8.VOL	<dt></dt>
2 REM	<ur></ur>	218 SOUND V2,P8+2,8,VOL:SOUND V3,RND(8)+	
3 REM	<us></us>	3,8,V0L	<xy></xy>
18 V0=0:V1=1:V2=2:V3=3:POKE 82,2:POKE 83		228 FOR P0=3 TO 12 STEP 0.02: VOL=P0/2:SO	
,39:GOTO 2000	<tx></tx>	UND V0,P0,8,VOL:SOUND V1,P0+1,8,VOL	<tc></tc>
50 POKE 540, WAIT	<yy></yy>	230 SOUND V2,P0+2,8,VOL:SOUND V3,RND(0) +	
52 IF PEEK (540) <>0 THEN 52	<00>	3,8,VOL:NEXT P8	<sk></sk>
54 FOR OFF=0 TO 3;SOUND OFF,0,0,0:NEXT O		240 FOR P0=10 TO 2 STEP -0.02:VOL-P0/2:S	
FF: RETURN	<ku></ku>	OUND V0,P0,8,VOL:SOUND V1,P0+1,8,VOL	<eb></eb>
89 REM	<kn></kn>	250 SOUND V2,P0+2,B,VOL:SOUND V3,RND(0) *	
90 REM *** MACHINE GUN ***	<qm></qm>	3,8,VOL:NEXT PØ:GOSUB 54:RETURN	<mb></mb>
91 REM	<b>&lt;</b> JY>	289 REM	<ca></ca>
100 FOR SHOT=1 TO 12:FOR VOL=15 TO 0 STE	- 1	290 REM *** LAZERS/PHOTONS ***	<c8></c8>
P -5: SOUND V0,80,0, VOL: SOUND V1,60,0, VOL	<va></va>	291 REM	<be></be>
110 SOUND V2,200,4,VDL:SOUND V3,10,4,VOL		300 FOR SHOT=1 TO 6:FOR P0=0 TO 200 STEP	
:NEXT VOL:GOSUB 54:NEXT SHOT	<gh></gh>	10	<en></en>
120 RETURN	<mp></mp>		
189 REM	<bz></bz>	Listing 5. Verschiedene Sound-Effekte. Bitte mit der	n
190 REM *** SURF/WAVES ***	<ty></ty>	Namen »D:SOUND.6« auf Diskette speichern.	



310 SOUND V0,P0,0,8:SOUND V1,P0,10,8:SOU	1	1200 FOR I=0 TO 3	<wl></wl>	
ND V2.P0.12.8:SOUND V3.P0.4.8	<aj></aj>		<90>	
329 NEXT PO:NEXT SHOT: GOSUB 54: RETURN	<\H>		<th></th>	
389 REM	<cb></cb>		<tl></tl>	
398 REM *** POLICE/FIRE SIREN ***	<de></de>		<l0></l0>	
	(BF)		<b0></b0>	
391 REN	COF /		<vk></vk>	
400 FOR P0=200 TO 50 STEP -1:SOUND V0.P0			<fk></fk>	
10,8:SOUND V1,P0+2,10,6:SOUND V2,P0+4,1	CTOS		<uh></uh>	
0,2:SOUND V3,P8+6,10,2:NEXT P0	<16>		<ix></ix>	
426 FOR P8-50 TO 160 STEP 6.2:50UND V0,P				
0,10,8:SOUND V1,P0+2,10,6:SOUND V2,P0+4,			<zy></zy>	
10,4:SOUND V3,P0+6,10,2:NEXT P8	<mt></mt>		<b6></b6>	
430 GOSUB 54: RETURN	<qz></qz>	1330 FOR P1=1 TO Y	<jz></jz>	
489 REM	<cc></cc>	1340 SOUND 0,P1,8,15	KEWD	
490 REM *** AIR RAID SIREN ***	<zx></zx>	1350 NEXT P1	<ez></ez>	
491 REM	<b6></b6>	1368 FOR DELAY=1 TO X:NEXT DELAY	<90>	
500 FOR LOOP=1 TO 6: FOR P0=1 TO 20: SOUND			<by></by>	
VØ,88+PØ,12,8:NEXT PØ	<kb></kb>		<vm></vm>	
510 SOUND V0,80,10,12:SOUND V1,100,10,12			<10>	
:SOUND V2,13,4,12	<ah></ah>		<u3></u3>	
520 WAIT=30:GOSUB 50:NEXT LOOP	<8X>		<tu></tu>	
530 FOR V=12 TO 0 STEP -0.1:SOUND V0.(20	100		<rr></rr>	
-V) +10,10,V:SOUND V1,(20-V)+10+20,10,V:S				
QUND V2,13,4,V:NEXT V	<dj></dj>		<hk></hk>	
540 GOSUB 54: RETURN	<rc></rc>		<kh></kh>	
589 REM	<cd></cd>		<u0></u0>	
			<nc></nc>	
590 REM +++ TELEPHONE RINGING +++	<bh></bh>		(UL>	
680 FOR RING=1 TO 2:FOR LOUD=1 TO 35:SOU			<hw></hw>	
			<nj></nj>	
ND V0,20,10,8:SOUND V1,1,2,0	<mn></mn>		<yr></yr>	
410 FOR LOOP=1 TO 2: SOUND V0,25,10,8: SOU			<60>	
ND V1,0,2,8:NEXT LOOP:SOUND V0,0,0,8:SOU			<cn></cn>	
ND V1,0,0,0:NEXT LOUD	<il></il>		<0X>	
620 FOR V=7 TO 8 STEP -8.2: SOUND V8.28,1			<ek></ek>	
0,V:SOUND VI,0,2,V:NEXT V	<cg></cg>	1570 NEXT I:NEXT X	<ha></ha>	
630 WAIT=90:GOSUB 50:NEXT RING:GOSUB 54:		1580 RETURN	<qb></qb>	
RETURN	<0H>	1999 STOP	<dw></dw>	
689 REM	<ce></ce>	2006 REM *** MENU OPTIONS ***	<xd></xd>	
690 REM *** WHISTLING BOMB ***	<ci></ci>	2010 GRAPHICS 0: SETCOLOR 2,15,0: POKE 752		
691 REM	<bi></bi>	,1:POKE 201,10:?	<en></en>	
700 FOR P0=0 TO 150:SOUND 0,P0,10,P0/15+		2020 7 ."(CTL Q) (CTL R) (CTL R) (CTL R) (CT		
2: NEXT PØ	<sa></sa>	L RECOTL RECOTL RECOTL RECOTL RECOTL REC		
718 FOR P0=0 TO 240 STEP 5: VOL=14-P0/20:		CTL R) (CTL R) (CTL R) (CTL R) (CTL R) (CTL R		
SOUND VØ,PØ,Ø,VOL:SOUND V1,PØ,9,VOL	<fc></fc>	}(CTL E)*	(VE)	
720 SOUND V2,P0+15,2,VOL:NEXT P0:60SUB 5		2030 ? ."(SHIFT =) SOUND EFFEKTE (SHIFT	***	
44 RETURN	<cc></cc>	=}"	(ED)	
789 REM	(CF)	2040 7 ." (CTL Z) (CTL R) (CTL R) (CTL R) (CTL	1207	
790 REM *** SPACE SHIP ***	(LF)	L R) (CTL R) (CTL R) (CTL R) (CTL R) (CTL R) (CTL R)		
791 REM	<bj></bj>	CTL RECTL RECTL RECTL RECTL RECTL RECTL RE		
800 SOUND V2,8,8,2:FOR VOL=1 TO 15 STEP	1.00		<8V>	
0.1:SOUND V0,25,4,VOL:SOUND V1,13,4,VOL:			<mj></mj>	
NEXT VOL	<sh></sh>		<ah></ah>	
810 FOR VOL=14 TO 8 STEP -8.1:SOUND V0.2				
5,4,VOL:SOUND V1,13,4,VOL:NEXT VOL	<gk></gk>		<00>	
826 GUSUB 54: RETURN	<rb></rb>		<re></re>	
889 REM	<c6></c6>		<mo></mo>	
			ZEVA	
NAM MEM 484 ADMIL FIRST THE		· ·	<fk></fk>	
890 REM *** SPACE ECHO ***	<vn></vn>	2150 ? "A<6>ATELEPHONE_RINGING"	<0N>	
891 REM		2150 ? "~<6>_TELEPHONE_RINGING" 2160 ? "~<7>_WHISTLING_BOMB"	<on></on>	
891 REM 986 FOR VOL-15 TO 8 STEP -0.2:FOR P8=0 T	<vn></vn>	2150 ? "A<6>ATELEPHONE RINGING" 2160 ? "A<7>AWHISTLING BOMB" 2170 ? "A<8>ASPACE ASHIP"	<on> <nn> <ry></ry></nn></on>	
891 REM 900 FOR VOL-15 TO 0 STEP -0.2:FOR P0=0 T 0 5:SOUND V0,P0,2,VOL:SOUND V1,P0+1,2,VO	<vn> <bk></bk></vn>	2150 ? "".<6>.TELEPHONE.RINGING" 2160 ? "".<7>.WHISTLING.BOMB" 2170 ? "".<8>.SPACE.SHIP" 2180 ? "".<9>.SPACE.ECHO"	<on> <nn> <ry> <be></be></ry></nn></on>	
891 REM 900 FOR VOL-15 TO 0 STEP -0.2:FOR P0=0 T 0 5:SOUND V0,P0,2,VOL:SOUND V1,P0+1,2,VO L:NEXT P0	<vn></vn>	2150 ? "A<6>ATELEPHONE RINGING" 2160 ? "A<7>AHHISTLING BOMB" 2170 ? "A<8>ASPACE ASHIP" 2180 ? "A<9>ASPACE ASHIP" 2190 ? "A<10>ADDOOR BELL"	<0N> <nn> <ry> <be> <bo></bo></be></ry></nn>	
891 REM 900 FOR VOL-15 TO 0 STEP -0.2:FOR P0=0 T 0 5:SOUND V0,P0,2,VOL:SOUND V1,P0+1,2,VO L:NEXT P0 910 FOR P1=VOL+10 TO VOL STEP -10:SOUND	<vn> <bk></bk></vn>	2150 ? "".<6>.TELEPHONE.RINGING" 2160 ? "".<7>.WHISTLING.BOMB" 2170 ? "".<8>.SPACE.SHIP" 2180 ? "".<9>.SPACE.ECHO" 2190 ? ".<10>.DOOR.BELL" 2200 ? "<11>.BUZZER"	<on> <nn> <nn> <ry> <be> <be> <um></um></be></be></ry></nn></nn></on>	
891 REM 900 FOR VOL-15 TO 0 STEP -0.2:FOR P0=0 T 0 5:SOUND V0,P0,2,VOL:SOUND V1,P0+1,2,VO L:NEXT P0 910 FOR P1=VOL+10 TO VOL STEP -10:SOUND V0,P1,10,VOL:SOUND V1,P1+VOL,10,VOL:NEXT	<uf></uf>	2150 ? ""<6>_TELEPHONE_RINGING" 2160 ? ""<7>_MHISTLING_BOMB" 2170 ? ""<8>_SPACE_SHIP" 2180 ? ""<9>_SPACE_ECHO" 2190 ? "<10>_DOOR_BELL" 2200 ? "<11>_BUZZER" 2210 ? "<12>_GALAXY_CANTINA_MUSIC"	<on> <nn> <nn> <ry> <be> <be> <um> <hm></hm></um></be></be></ry></nn></nn></on>	
891 REM 900 FOR VOL-15 TO 0 STEP -0.2:FOR P0=0 T 0 5:SOUND V0,P0,2,VOL:SOUND V1,P0+1,2,VO L:NEXT P0 910 FOR P1=VOL+10 TO VOL STEP -10:SOUND V0,P1,10,VOL:SOUND V1,P1+VOL,10,VOL:NEXT P1:NEXT VOL	<pre><vn></vn></pre>	2150 ? "".<6>.TELEPHONE.RINGING" 2160 ? "".<7>.WHISTLING.BOMB" 2170 ? "".<8>.SPACE.SHIP" 2180 ? "".<9>.SPACE.ECHO" 2190 ? ".<10>.DOOR.BELL" 2200 ? ".<11>.BUZZER" 2210 ? ".<12>.GALAXY.CANTINA_MUSIC" 2220 ? ".<13>.THUNDER"	<on> <nn> <nn> <ry> <be> <be> <bo> <um> <hm> <rg></rg></hm></um></bo></be></be></ry></nn></nn></on>	
891 REM 900 FOR VOL-15 TO 0 STEP -0.2:FOR P0=0 T 0 5:SOUND V0,P0,2,VOL:SOUND V1,P0+1,2,VO L:NEXT P0 910 FOR P1=VOL*10 TO VOL STEP -10:SOUND V0,P1,10,VOL:SOUND V1,P1+VOL,10,VOL:NEXT P1:NEXT VOL 920 RETURN	<un> <bk> <uf> <uf> &lt;</uf></uf></bk></un>	2150 ? ""<6>_TELEPHONE_RINGING"  2160 ? ""<7>_WHISTLING_BOMB"  2170 ? ""<8>_SPACE_SHIP"  2180 ? ""<9>_SPACE_ECHO"  2190 ? "<10>_DOOR_BELL"  2200 ? "<11>_BUZZER"  2210 ? "<12>_GALAXY_CANTINA_MUSIC"  2220 ? "<13>_THUNDER"  2230 ? "<14>_DU_AIRPLANE"	<on> <nn> <nn> <ry> <be> <be> <um> <hm></hm></um></be></be></ry></nn></nn></on>	
891 REM 900 FOR VOL-15 TO 0 STEP -0.2:FOR P0=0 T 0 5:SOUND V0,P0,2,VOL:SOUND V1,P0+1,2,VO L:NEXT P0 910 FOR P1=VOL*10 TO VOL STEP -10:SOUND V0,P1,10,VOL:SOUND V1,P1+VOL,10,VOL:NEXT P1:NEXT VOL 920 RETURN 989 REM	<pre></pre> <pre>&lt;</pre>	2150 ? ""<6>_TELEPHONE_RINGING" 2160 ? ""<7>_WHISTLING_BOMB" 2170 ? ""<8>_SPACE_SHIP" 2180 ? ""<9>_SPACE_ECHO" 2190 ? "<10>_DOOR_BELL" 2200 ? "<11>_BUZZER" 2210 ? "<12>_GALAXY_CANTINA_MUSIC" 2220 ? "<13>_THUNDER" 2230 ? "<14>_DOLD_AIRPLANE" 2240 ? "<15>_PLANE_CRASHING"	<pre><on> <nn> <nn> <ry> <be> <bo> <um> <hw> <hw> <rg> <um> <hw> <hw> <iun <iun="" <iun<="" td=""></iun></hw></hw></um></rg></hw></hw></um></bo></be></ry></nn></nn></on></pre>	
891 REM 900 FOR VOL-15 TO 0 STEP -0.2:FOR P0=0 T 0 5:SOUND V0,P0,2,VOL:SOUND V1,P0+1,2,VO L:NEXT P0 910 FOR P1=VOL*10 TO VOL STEP -10:SOUND V0,P1,10,VOL:SOUND V1,P1+VOL,10,VOL:NEXT P1:NEXT VOL 920 RETURN 989 REM 990 REM ### DOOR BELL ###	CVN> CBC>  CUF>  CDG> CHC> CCH> CTC>	2150 ? "".<6>.TELEPHONE.RINGING" 2160 ? "".<7>.WHISTLING.BOMB" 2170 ? "".<8>.SPACE.SHIP" 2180 ? "".<9>.SPACE.ECHO" 2190 ? ""<10>.DOOR.BELL" 2200 ? "<11>.BUZZER" 2210 ? "<12>.GALAXY.CANTINA.MUSIC" 2220 ? "<13>.THUNDER" 2230 ? "<14>.DU.D.AIRPLANE" 2240 ? "<15>.PLANE.CRASHING" 2500 ? "<16>.RUN.LAST.PROGRAM"	<pre><on> <nn> <nn> <ry> <be> <be> <um> <um> <hw> <rg> <iu> </iu></rg></hw></um></um></be></be></ry></nn></nn></on></pre>	
891 REM 900 FOR VOL-15 TO 0 STEP -0.2:FOR P0=0 T 0 5:SOUND V0,P0,2,VOL:SOUND V1,P0+1,2,VO L:NEXT P0 910 FOR P1=VOL*10 TO VOL STEP -10:SOUND V0,P1,10,VOL:SOUND V1,P1+VOL,10,VOL:NEXT P1:NEXT VOL 920 RETURN 989 REM 990 REM *** DOOR BELL ***	<pre></pre> <pre>&lt;</pre>	2150 ? "".<6>.TELEPHONE.RINGING" 2160 ? "".<7>.WHISTLING.BOMB" 2170 ? "".<8>.SPACE.SHIP" 2180 ? "".<9>.SPACE.ECHO" 2190 ? "<10>.DOOR.BELL" 2200 ? "<11>.BUZZER" 2210 ? "<12>.GALAXY.CANTINA.MUSIC" 2220 ? "<13>.THUNDER" 2230 ? "<14>.DU.D.AIRPLANE" 2240 ? "<15>.PLANE.CRASHING" 2500 ? "<16>.RUNALAST.PROGRAM" 3000 POKE 53279,0:? :? ""	<pre><on> <nn> <nn> <ry> <be> <bo> <um> <hw> <hw> <rg> <um> <hw> <hw> <iun <iun="" <iun<="" td=""></iun></hw></hw></um></rg></hw></hw></um></bo></be></ry></nn></nn></on></pre>	
891 REM 900 FOR VOL-15 TO 0 STEP -0.2:FOR P0=0 T 0 5:SOUND V0.P0.2,VOL:SOUND V1.P0+1.2,VO L:NEXT P0 910 FOR P1=VOL*10 TO VOL STEP -10:SOUND V0.P1.10.VOL:SOUND V1.P1+VOL.10.VOL:NEXT P1:NEXT VOL 920 RETURN 989 REM 990 REM *** DOOR BELL *** 991 REM 1000 FOR VOL=15 TO 0 STEP -0.5:SOUND V0.	<ur> <bk> <uf> <uf> <amount c<="" control="" of="" td="" the=""><td>2150 ? "".&lt;6&gt;.TELEPHONE_RINGING" 2160 ? "".&lt;7&gt;.WHISTLING_BOMB" 2170 ? "".&lt;8&gt;.SPACE_SHIP" 2180 ? "".&lt;9&gt;.SPACE_ECHO" 2190 ? ""&lt;10&gt;.DOOR_BELL" 2200 ? "&lt;11&gt;.BUZZER" 2210 ? "&lt;12&gt;.GALAXY_DANTINA_MUSIC" 2228 ? "&lt;13&gt;.THUNDER" 2230 ? "&lt;14&gt;.OUD_AIRPLANE" 2240 ? "&lt;15&gt;.PLANE_CRASHING" 2500 ? "&lt;16&gt;.RUN_LAST_PROGRAM" 3000 POKE 53279,0:? :? ""WELCHE_NR." ";:TRAP 9000:INPUT CHOICE:TRAP 40000</td><td><pre><on> <nn> <nn> <ry> <be> <bo> <um> <hw> <hw> <rg> <um> <hw> <hw> <iun <iun="" <iun<="" td=""></iun></hw></hw></um></rg></hw></hw></um></bo></be></ry></nn></nn></on></pre></td></amount></uf></uf></bk></ur>	2150 ? "".<6>.TELEPHONE_RINGING" 2160 ? "".<7>.WHISTLING_BOMB" 2170 ? "".<8>.SPACE_SHIP" 2180 ? "".<9>.SPACE_ECHO" 2190 ? ""<10>.DOOR_BELL" 2200 ? "<11>.BUZZER" 2210 ? "<12>.GALAXY_DANTINA_MUSIC" 2228 ? "<13>.THUNDER" 2230 ? "<14>.OUD_AIRPLANE" 2240 ? "<15>.PLANE_CRASHING" 2500 ? "<16>.RUN_LAST_PROGRAM" 3000 POKE 53279,0:? :? ""WELCHE_NR." ";:TRAP 9000:INPUT CHOICE:TRAP 40000	<pre><on> <nn> <nn> <ry> <be> <bo> <um> <hw> <hw> <rg> <um> <hw> <hw> <iun <iun="" <iun<="" td=""></iun></hw></hw></um></rg></hw></hw></um></bo></be></ry></nn></nn></on></pre>	
891 REM 900 FOR VOL-15 TO 0 STEP -0.2:FOR P0=0 T 0 5:SOUND V0,P0,2,VOL:SOUND V1,P0+1,2,VO L:NEXT P0 910 FOR P1=VOL*10 TO VOL STEP -10:SOUND V0,P1,10,VOL:SOUND V1,P1+VOL,10,VOL:NEXT P1:NEXT VOL 920 RETURN 989 REM 990 REM *** DOOR BELL *** 991 REM 1000 FOR VOL=15 TO 0 STEP -0.5:SOUND V0, 29,10,VOL:SOUND V1,30,10,VOL:NEXT VOL	CVN> CBC>  CUF>  CDG> CHC> CCH> CTC>	2150 ? "".<6>.TELEPHONE.RINGING" 2160 ? "".<7>.WHISTLING.BOMB" 2170 ? "".<8>.SPACE.SHIP" 2180 ? "".<9>.SPACE.ECHO" 2190 ? "<16>.DOOR.BELL" 2200 ? "<11>.BUZZER" 2210 ? "<12>.GALAXY.DANTINA.MUSIC" 2220 ? "<13>.THUNDER" 2230 ? "<14>.OLD.AIRPLANE" 2240 ? "<15>.PLANE.CRASTING" 2500 ? "<16>.RUN.LAST.PROGRAM" 3000 POKE 53279,0:? :? ""WELCHE.NR." ";:TRAP 9000:INPUT CHOICE:TRAP 40000 3010 CHOICE=INT(CHOICE):IF CHOICE<1 OR C	<pre><on> <nn> <nn> <ry> <be> <bo> <um> <hw> <rg> <iu> <jz> <iu> <jiii <="" pre=""></jiii></iu></jz></iu></rg></hw></um></bo></be></ry></nn></nn></on></pre>	
891 REM 900 FOR VOL-15 TO 0 STEP -0.2:FOR P0=0 T 0 5:SOUND V0,P0,2,VOL:SOUND V1,P0+1,2,VO L:NEXT P0 910 FOR P1=VOL*10 TO VOL STEP -10:SOUND V0,P1,10,VOL:SOUND V1,P1+VOL,10,VOL:NEXT P1:NEXT VOL 920 RETURN 989 REM 990 REM *** DOOR BELL *** 991 REM 1000 FOR VOL=15 TO 0 STEP -0.5:SOUND V0, 29,10,VOL:SOUND V1,30,10,VOL:NEXT VOL 1010 FOR VOL=15 TO 0 STEP -0.5:SOUND V0,	<un> <bk> <uf> <uf> <cog> <hx> <ch> <tic> <bl> <fs></fs></bl></tic></ch></hx></cog></uf></uf></bk></un>	2150 ? "".<6>.TELEPHONE.RINGING" 2160 ? "".<7>.WHISTLING.BOMB" 2170 ? "".<8>.SPACE.SHIP" 2180 ? "".<9>.SPACE.ECHO" 2190 ? "<10>.DOOR.BELL" 2200 ? "<11>.BUZZER" 2210 ? "<12>.GALAXY.CANTINA.MUSIC" 2220 ? "<13>.THUNDER" 2230 ? "<14>.OLD.AIRPLANE" 2240 ? "<15>.PLANE.CRASHING" 2500 ? "<16>.RUN.LASTPROGRAM" 3000 POKE 53279,0:? :? ""	<pre><on> <nn> <nn> <ry> <be> <be> <bd> <um> <hm> <rg> <um> <hm> <rg> <iu> <iii <ii="" <iii="" i<="" ii="" td=""></iii></iu></rg></hm></um></rg></hm></um></bd></be></be></ry></nn></nn></on></pre>	
891 REM 900 FOR VOL-15 TO 0 STEP -0.2:FOR P0=0 T 0 5:SOUND V0,P0,2,VOL:SOUND V1,P0+1,2,VO L:NEXT P0 910 FOR P1=VOL*10 TO VOL STEP -10:SOUND V0,P1,10,VOL:SOUND V1,P1+VOL,10,VOL:NEXT P1:NEXT VOL 920 RETURN 989 REM 990 REM *** DOOR BELL *** 991 REM 1000 FOR VOL=15 TO 0 STEP -0.5:SOUND V0, 29,10,VOL:SOUND V1,30,10,VOL:NEXT VOL 1010 FOR VOL=15 TO 0 STEP -0.5:SOUND V0, 35,10,VOL:SOUND V1,36,10,VOL:NEXT VOL	<un> <bk> <uf> <uf> <amount control="" of="" td="" the="" the<=""><td>2150 ? "".&lt;6&gt;.TELEPHONE_RINGING" 2160 ? "".&lt;7&gt;.WHISTLING_BOMB" 2170 ? "".&lt;8&gt;.SPACE_SHIP" 2180 ? "".&lt;9&gt;.SPACE_ECHO" 2190 ? ".&lt;10&gt;.DOOR_BELL" 2200 ? "&lt;11&gt;.BUZZER" 2210 ? "&lt;12&gt;.GALAXY_CANTINA_MUSIC" 2220 ? "&lt;13&gt;.THUNDER" 2230 ? "&lt;14&gt;.OLD_AIRPLANE" 2240 ? "&lt;15&gt;.PLANE_CRASHING" 2500 ? "&lt;16&gt;.RUN_LAST_PROGRAM" 3000 POKE 5327,0:? :? ""</td><td><pre><on> <nn> <nn> <ry> <be> <be> <um> <hw> <hw> <chu <the="" <the<="" td=""></chu></hw></hw></um></be></be></ry></nn></nn></on></pre></td></amount></uf></uf></bk></un>	2150 ? "".<6>.TELEPHONE_RINGING" 2160 ? "".<7>.WHISTLING_BOMB" 2170 ? "".<8>.SPACE_SHIP" 2180 ? "".<9>.SPACE_ECHO" 2190 ? ".<10>.DOOR_BELL" 2200 ? "<11>.BUZZER" 2210 ? "<12>.GALAXY_CANTINA_MUSIC" 2220 ? "<13>.THUNDER" 2230 ? "<14>.OLD_AIRPLANE" 2240 ? "<15>.PLANE_CRASHING" 2500 ? "<16>.RUN_LAST_PROGRAM" 3000 POKE 5327,0:? :? ""	<pre><on> <nn> <nn> <ry> <be> <be> <um> <hw> <hw> <chu <the="" <the<="" td=""></chu></hw></hw></um></be></be></ry></nn></nn></on></pre>	
891 REM 900 FOR VOL-15 TO 0 STEP -0.2:FOR P0=0 T 0 5:SOUND V0,P0,2,VOL:SOUND V1,P0+1,2,VO L:NEXT P0 910 FOR P1=VOL*10 TO VOL STEP -10:SOUND V0,P1,10,VOL:SOUND V1,P1+VOL,10,VOL:NEXT P1:NEXT VOL 920 RETURN 989 REM 990 REM *** DOOR BELL *** 791 REM 1000 FOR VOL=15 TO 0 STEP -0.5:SOUND V0, 29,10,VOL:SOUND V1,30,10,VOL:NEXT VOL 1010 FOR VOL=15 TO 0 STEP -0.5:SOUND V0, 35,10,VOL:SOUND V1,36,10,VOL:NEXT VOL 1020 RETURN	<pre></pre>	2150 ? "".<6>.TELEPHONE_RINGING" 2160 ? "".<7>.WHISTLING_BOMB" 2170 ? "".<8>.SPACE_SHIP" 2180 ? "".<9>.SPACE_ECHO" 2190 ? "<10>.DOOR_BELL" 2200 ? "<11>.BUZZER" 2210 ? "<11>.BUZZER" 2210 ? "<12>.GALAXY_CANTINA_MUSIC" 2220 ? "<13>.THUNDER" 2230 ? "<14>.OLD_AIRPLANE" 2240 ? "<15>.PLANE_CRASHING" 2500 ? "<16>.RUN_LAST_PROGRAM" 3000 POKE 53279,0:? :? ""	<pre><on> <nn> <nn> <ry> <be> <be> <bd> <um> <nm> <nm> <nm> <nm> <um> <nm> <nm> <um> <nm> <um> <nm> <um> <nm> <um> <um> <um> <um> <um> <um> <um> <u< td=""></u<></um></um></um></um></um></um></um></nm></um></nm></um></nm></um></nm></nm></um></nm></nm></nm></nm></um></bd></be></be></ry></nn></nn></on></pre>	
891 REM 900 FOR VOL-15 TO 0 STEP -0.2:FOR P0=0 T 0 5:SOUND V0,P0,2,VOL:SOUND V1,P0+1,2,VO L:NEXT P0 910 FOR P1=VOL*10 TO VOL STEP -10:SOUND V0,P1,10,VOL:SOUND V1,P1+VOL,10,VOL:NEXT P1:NEXT VOL 920 RETURN 989 REM 990 REM *** DOOR BELL *** 991 REM 1000 FOR VOL=15 TO 0 STEP -0.5:SOUND V0, 29,10,VOL:SOUND V1,30,10,VOL:NEXT VOL 1010 FOR VOL=15 TO 0 STEP -0.5:SOUND V0, 35,10,VOL:SOUND V1,36,10,VOL:NEXT VOL 1020 RETURN 1089 REM	<pre></pre>	2150 ? "".<6>.TELEPHONE_RINGING" 2160 ? "".<7>.WHISTLING_BOMB" 2170 ? "".<8>.SPACE_SHIP" 2190 ? ".<9>.SPACE_ECHO" 2190 ? "<10>.DOOR_BELL" 2200 ? "<11>.BUZZER" 2210 ? "<11>.BUZZER" 2220 ? "<13>.THUNDER" 2230 ? "<14>.DLD_AIRPLANE" 2240 ? "<15>.PLANE_CRASHING" 2500 ? "<16>.RUN_LAST_PROGRAM" 3000 POKE 53279,0:? :? ""	<pre><on> <nn> <nn> <ry> <be> <be> <bd> <um> <nm> <nm> <nm> <nm> <um> <nm> <nm> <um> <nm> <um> <nm> <um> <nm> <um> <um> <um> <um> <um> <um> <um> <u< td=""></u<></um></um></um></um></um></um></um></nm></um></nm></um></nm></um></nm></nm></um></nm></nm></nm></nm></um></bd></be></be></ry></nn></nn></on></pre>	
891 REM 900 FOR VOL-15 TO 0 STEP -0.2:FOR P0=0 T 0 5:SOUND V0,P0,2,VOL:SOUND V1,P0+1,2,VO L:NEXT P0 910 FOR P1=VOL*10 TO VOL STEP -10:SOUND V0,P1,10,VOL:SOUND V1,P1+VOL,10,VOL:NEXT P1:NEXT VOL 920 RETURN 989 REM 990 REM *** DOOR BELL *** 991 REM 1000 FOR VOL=15 TO 0 STEP -0.5:SOUND V0, 29,10,VOL:SOUND V1,30,10,VOL:NEXT VOL 1010 FOR VOL=15 TO 0 STEP -0.5:SOUND V0, 35,10,VOL:SOUND V1,36,10,VOL:NEXT VOL 1020 RETURN 1089 REM 1090 REM *** BUZZER ***	<un> <bk> <uf> <uf> <amount c<="" control="" of="" td="" the=""><td>2150 ? "".&lt;6&gt;.TELEPHONE_RINGING" 2160 ? "".&lt;7&gt;.WHISTLING_BOMB" 2170 ? "".&lt;8&gt;.SPACE_SHIP" 2180 ? "".&lt;9&gt;.SPACE_ECHO" 2190 ? ".&lt;10&gt;.DOUR_BELL" 2200 ? "&lt;11&gt;.BUZZER" 2210 ? "&lt;12&gt;.GALAXY_DANTINA_MUSIC" 2228 ? "&lt;13&gt;.THUNDER" 2230 ? "&lt;14&gt;.DUD_AIRPLANE" 2240 ? "&lt;15&gt;.PLANE_CRASHING" 2500 ? "&lt;16&gt;.RUN_LAST_PROGRAM" 3000 POKE 53279,0:? :? ""</td><td><pre><on> <nn> <nn> <ry> <be> <be> <um> <nw> <rg> <um> <nw> <rg> <iun <iun="" <iun<="" td=""></iun></rg></nw></um></rg></nw></um></be></be></ry></nn></nn></on></pre></td></amount></uf></uf></bk></un>	2150 ? "".<6>.TELEPHONE_RINGING" 2160 ? "".<7>.WHISTLING_BOMB" 2170 ? "".<8>.SPACE_SHIP" 2180 ? "".<9>.SPACE_ECHO" 2190 ? ".<10>.DOUR_BELL" 2200 ? "<11>.BUZZER" 2210 ? "<12>.GALAXY_DANTINA_MUSIC" 2228 ? "<13>.THUNDER" 2230 ? "<14>.DUD_AIRPLANE" 2240 ? "<15>.PLANE_CRASHING" 2500 ? "<16>.RUN_LAST_PROGRAM" 3000 POKE 53279,0:? :? ""	<pre><on> <nn> <nn> <ry> <be> <be> <um> <nw> <rg> <um> <nw> <rg> <iun <iun="" <iun<="" td=""></iun></rg></nw></um></rg></nw></um></be></be></ry></nn></nn></on></pre>	
891 REM 900 FOR VOL-15 TO 0 STEP -0.2:FOR P0=0 T 0 5:SOUND V0,P0,2,VOL:SOUND V1,P0+1,2,VO L:NEXT P0 910 FOR P1=VOL*10 TO VOL STEP -10:SOUND V2,P1,10,VOL:SOUND V1,P1+VOL,10,VOL:NEXT P1:NEXT VOL 920 RETURN 989 REM 990 REM *** DOOR BELL *** 991 REM 1000 FOR VOL=15 TO 0 STEP -0.5:SOUND V0, 29,10,VOL:SOUND V1,30,10,VOL:NEXT VOL 1010 FOR VOL=15 TO 0 STEP -0.5:SOUND V0, 35,10,VOL:SOUND V1,36,10,VOL:NEXT VOL 1020 RETURN 1089 REM 1090 REM *** BUZZER ***	<pre><vn> <bk>  <uf>  <uf>  <d6> <mx> <chc <tic=""> <tic> <bl> </bl></tic></chc></mx></d6></uf></uf></bk></vn></pre> <pre><pa> <oz> <ma> <ub> </ub></ma></oz></pa></pre>	2150 ? "".<6>.TELEPHONE.RINGING" 2160 ? "".<7>.WHISTLING.BOMB" 2170 ? "".<8>.SPACE.SHIP" 2180 ? ".<9>.SPACE.ECHO" 2190 ? "<10>.DOOR.BELL" 2200 ? "<11>.BUZZER" 2210 ? "<12>.GALAXY.DANTINA.MUSIC" 2220 ? "<13>.THUNDER" 2230 ? "<14>.OLD.AIRPLANE" 2240 ? "<15>.PLANE.CRASTING" 2500 ? "<16>.RUN.LAST.PROGRAM" 3000 POKE 53279,0:? :? ""	<pre><on> <nn> <nn> <ry> <bc> <bc> <um> <hw> <rg> <iu> <iii <ii="" <iii="" ii="" ii<="" td=""></iii></iu></rg></hw></um></bc></bc></ry></nn></nn></on></pre>	
891 REM 906 FOR VOL-15 TO 0 STEP -0.2:FOR P0=0 T 0 5:SOUND V0,P0,2,VOL:SOUND V1,P0+1,2,VO L:NEXT P0 910 FOR P1=VOL*10 TO VOL STEP -10:SOUND V0,P1,10,VOL:SOUND V1,P1+VOL,10,VOL:NEXT P1:NEXT VOL 920 RETURN 989 REM 990 REM *** DOOR BELL *** 991 REM 1000 FOR VOL=15 TO 0 STEP -0.5:SOUND V0, 29,10,VOL:SOUND V1,30,10,VOL:NEXT VOL 1010 FOR VOL=15 TO 0 STEP -0.5:SOUND V0, 35,10,VOL:SOUND V1,36,10,VOL:NEXT VOL 1020 RETURN 1089 REM 1070 REM *** BUZZER *** 1091 REM 1100 SOUND 0,40,6,10	<pre></pre> <pre><uf> <uf> <uf> </uf></uf></uf></pre> <pre><uf> </uf></pre> <pre><pre><ue> </ue></pre> <pre><ue> <pre><ue< pre=""> <pre><ue< p=""> <pre><ue< pre=""> <pre><ue< p=""> <pre><ue< pre=""> <ue< p=""> <pre><ue< p=""> <pre><ue< p=""> <ue< p=""> <pre><ue< p=""> <ue< p=""> <pre><ue< p=""> <ue< p=""> <pre><ue< p=""> <ue< p=""> <p< td=""><td>2150 ? "".&lt;6&gt;.TELEPHONE.RINGING" 2160 ? "".&lt;7&gt;.WHISTLING.BOMB" 2170 ? "".&lt;8&gt;.SPACE.SHIP" 2180 ? "".&lt;9&gt;.SPACE.ECHO" 2190 ? "&lt;10&gt;.DOOR.BELL" 2200 ? "&lt;11&gt;.BUZZER" 2210 ? "&lt;12&gt;.GALAXY.CANTINA.MUSIC" 2220 ? "&lt;13&gt;.THUNDER" 2230 ? "&lt;14&gt;.OLD.AIRPLANE" 2240 ? "&lt;15&gt;.PLANE.CRASHING" 2580 ? "&lt;16&gt;.RUNALAST.PROGRAM" 3000 POKE 53279,0:? :? ""AAA.WELCHE.NR." ":TRAP 9000:INPUT CHOICE:TRAP 40000 3010 CHOICE=INT(CHOICE):IF CHOICE&lt;1 OR CHOICE&gt;16 THEN 9000 3020 IF CHOICE=16 THEN RUN "D:SOUND.7" 3030 GRAPHICS 0:SETCOLOR 2,CHOICE,0:? :? :? :LIST CHOICE*100-10,CHOICE*100+80 3046 GOSUB CHOICE*100:? :? "ADURECK EAEINE.TASTE.FUER.MENUE";:POKE 764,255 3050 IF PEEK(764)=255 THEN 3050</td><td><pre><on> <nn> <nn> <ry> <be> <be> <bd> <um> <nm> <rg> <um> <nm> <nm> <rg> <iu> <cbc <ii="" <iii="" <iu="" ii="" ii<="" td=""></cbc></iu></rg></nm></nm></um></rg></nm></um></bd></be></be></ry></nn></nn></on></pre></td></p<></ue<></ue<></ue<></ue<></ue<></ue<></ue<></ue<></ue<></ue<></ue<></ue<></ue<></ue<></ue<></ue<></ue<></ue<></ue<></ue<></ue<></ue<></ue<></ue<></ue<></ue<></pre></ue<></ue<></pre></ue<></ue<></pre></ue<></ue<></pre></ue<></pre></ue<></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue<></pre></ue></pre></pre>	2150 ? "".<6>.TELEPHONE.RINGING" 2160 ? "".<7>.WHISTLING.BOMB" 2170 ? "".<8>.SPACE.SHIP" 2180 ? "".<9>.SPACE.ECHO" 2190 ? "<10>.DOOR.BELL" 2200 ? "<11>.BUZZER" 2210 ? "<12>.GALAXY.CANTINA.MUSIC" 2220 ? "<13>.THUNDER" 2230 ? "<14>.OLD.AIRPLANE" 2240 ? "<15>.PLANE.CRASHING" 2580 ? "<16>.RUNALAST.PROGRAM" 3000 POKE 53279,0:? :? ""AAA.WELCHE.NR." ":TRAP 9000:INPUT CHOICE:TRAP 40000 3010 CHOICE=INT(CHOICE):IF CHOICE<1 OR CHOICE>16 THEN 9000 3020 IF CHOICE=16 THEN RUN "D:SOUND.7" 3030 GRAPHICS 0:SETCOLOR 2,CHOICE,0:? :? :? :LIST CHOICE*100-10,CHOICE*100+80 3046 GOSUB CHOICE*100:? :? "ADURECK EAEINE.TASTE.FUER.MENUE";:POKE 764,255 3050 IF PEEK(764)=255 THEN 3050	<pre><on> <nn> <nn> <ry> <be> <be> <bd> <um> <nm> <rg> <um> <nm> <nm> <rg> <iu> <cbc <ii="" <iii="" <iu="" ii="" ii<="" td=""></cbc></iu></rg></nm></nm></um></rg></nm></um></bd></be></be></ry></nn></nn></on></pre>	
891 REM 906 FOR VOL-15 TO 0 STEP -0.2:FOR P0=0 T 0 5:SOUND V0,P0,2,VOL:SOUND V1,P0+1,2,VO L:NEXT P0 910 FOR P1=VOL*10 TO VOL STEP -10:SOUND V0,P1,10,VOL:SOUND V1,P1+VOL,10,VOL:NEXT P1:NEXT VOL 920 RETURN 989 REM 990 REM *** DOOR BELL *** 991 REM 1000 FOR VOL=15 TO 0 STEP -0.5:SOUND V0, 27,10,VOL:SOUND V1,30,10,VOL:NEXT VOL 1010 FOR VOL=15 TO 0 STEP -0.5:SOUND V0, 35,10,VOL:SOUND V1,36,10,VOL:NEXT VOL 1020 RETURN 1089 REM 1090 REM *** BUZZER *** 1091 REM 1100 SOUND 0,40,6,10 1110 FOR DELAY-1 TO 400	<pre></pre> <pre><uf> <uf> <uf> </uf></uf></uf></pre> <pre><uf> </uf></pre> <pre><pre><pre><pre><pre><pre><pre><pre></pre></pre></pre></pre></pre></pre></pre></pre>	2150 ? ""<6>TELEPHONE_RINGING" 2160 ? ""<7>MHISTLING_BOMB" 2170 ? ""<8>SPACE_SHIP" 2180 ? ""<9>SPACE_ECHO" 2190 ? "<10>DOOR_BELL" 2200 ? "<11>BUZZER" 2210 ? "<12>GALAXY_CANTINA_MUSIC" 2220 ? "<13>THUNDER" 2230 ? "<14>COLD_AIRPLANE" 2240 ? "<15>PLANE_CRASHING" 2500 ? "<16>RUNALAST_PROGRAM" 3000 POKE 53279,0:? :? ""AAA_WELCHE_NR." ";:TRAP 9000:INPUT CHOICE:TRAP 40000 3010 CHOICE=INT(CHOICE):IF CHOICE<1 OR CHOICE>16 THEN 9000 3020 IF CHOICE=16 THEN RUN "D:SOUND.7" 3030 GRAPHICS 0:SETCOLOR 2,CHOICE,0:? :? :? :? :LIST CHOICE*100-10,CHOICE*100+80 3040 GOSUB CHOICE*100? :? ""AADURECK E_EINE_TASTE_FUER_MENUE";:POKE 764,255 3060 POKE 764,255:BOTO 2000	<pre><on> <nn> <nn> <ry> <be> <be> <bd> <um> <nm> <rg> <um <nm=""> <nm> <iun <rg=""> <iun <inn="" <inn<="" <iun="" td=""></iun></iun></nm></um></rg></nm></um></bd></be></be></ry></nn></nn></on></pre>	
891 REM 900 FOR VOL-15 TO 0 STEP -0.2:FOR P0=0 T 0 5:SOUND V0,P0,2,VOL:SOUND V1,P0+1,2,VO L:NEXT P0 910 FOR P1=VOL*10 TO VOL STEP -10:SOUND V0,P1,10,VOL:SOUND V1,P1+VOL,10,VOL:NEXT P1:NEXT VOL 920 RETURN 989 REM 990 REM *** DOOR BELL *** 991 REM 1000 FOR VOL=15 TO 0 STEP -0.5:SOUND V0, 29,10,VOL:SOUND V1,30,10,VOL:NEXT VOL 1010 FOR VOL=15 TO 0 STEP -0.5:SOUND V0, 35,10,VOL:SOUND V1,36,10,VOL:NEXT VOL 1020 RETURN 1089 REM 1090 REM *** BUZZER *** 1091 REM 1100 SOUND 0,40,6,10 1110 FOR DELAY-1 TO 400 1120 NEXT DELAY	<pre><vn> <bk>  <uf>  <uf>  <uf>  <uf> </uf></uf></uf></uf></bk></vn></pre> <pre><mx> <ch> <tk> <bl> </bl></tk></ch></mx></pre> <pre><vb> <ub> <ub> <ub> <ub> <ub> <ub> <ub> <u< td=""><td>2150 ? "".&lt;6&gt;.TELEPHONE_RINGING" 2160 ? "".&lt;7&gt;.WHISTLING_BOMB" 2170 ? "".&lt;8&gt;.SPACE_SHIP" 2190 ? "".&lt;9&gt;.SPACE_ECHO" 2190 ? "&lt;10&gt;.DOOR_BELL" 2200 ? "&lt;11&gt;.BUZZER" 2210 ? "&lt;11&gt;.BUZZER" 2210 ? "&lt;12&gt;.GALAXY_CANTINA_MUSIC" 2220 ? "&lt;13&gt;.THUNDER" 2230 ? "&lt;14&gt;.OLD_AIRPLANE" 2240 ? "&lt;15&gt;.PLANE_CRASHING" 2500 ? "&lt;16&gt;.RUN_LAST_PROGRAM" 3000 POKE 5327,0:? :? ""WELCHE_NR." ":TRAP 9000:INPUT CHOICE:TRAP 40000 3010 CHOICE=INT(CHOICE):IF CHOICE&lt;1 OR CHOICE&gt;16 THEN 9000 3020 IF CHOICE=16 THEN RUN "D:SOUND.7" 3030 GRAPHICS 0:SETCOLOR 2,CHOICE,0:? :? :? :LIST CHOICE*100-10,CHOICE*100+80 3040 GOSUB CHOICE*100:? :? "DURECK E_EINE_TASTE_FUER_MENUE";:POKE 764,255 3050 IF PEEK (764)=255 THEN 3050 3060 POKE 764,255:BOTO 2000</td><td><pre><on> <nn> <nn> <ry> <be> <be> <bd> <um> <nm> <rg> <um> <nm> <nm> <rg> <iu> <cbc <ii="" <iii="" <iu="" ii="" ii<="" td=""></cbc></iu></rg></nm></nm></um></rg></nm></um></bd></be></be></ry></nn></nn></on></pre></td></u<></ub></ub></ub></ub></ub></ub></ub></vb></pre>	2150 ? "".<6>.TELEPHONE_RINGING" 2160 ? "".<7>.WHISTLING_BOMB" 2170 ? "".<8>.SPACE_SHIP" 2190 ? "".<9>.SPACE_ECHO" 2190 ? "<10>.DOOR_BELL" 2200 ? "<11>.BUZZER" 2210 ? "<11>.BUZZER" 2210 ? "<12>.GALAXY_CANTINA_MUSIC" 2220 ? "<13>.THUNDER" 2230 ? "<14>.OLD_AIRPLANE" 2240 ? "<15>.PLANE_CRASHING" 2500 ? "<16>.RUN_LAST_PROGRAM" 3000 POKE 5327,0:? :? ""WELCHE_NR." ":TRAP 9000:INPUT CHOICE:TRAP 40000 3010 CHOICE=INT(CHOICE):IF CHOICE<1 OR CHOICE>16 THEN 9000 3020 IF CHOICE=16 THEN RUN "D:SOUND.7" 3030 GRAPHICS 0:SETCOLOR 2,CHOICE,0:? :? :? :LIST CHOICE*100-10,CHOICE*100+80 3040 GOSUB CHOICE*100:? :? "DURECK E_EINE_TASTE_FUER_MENUE";:POKE 764,255 3050 IF PEEK (764)=255 THEN 3050 3060 POKE 764,255:BOTO 2000	<pre><on> <nn> <nn> <ry> <be> <be> <bd> <um> <nm> <rg> <um> <nm> <nm> <rg> <iu> <cbc <ii="" <iii="" <iu="" ii="" ii<="" td=""></cbc></iu></rg></nm></nm></um></rg></nm></um></bd></be></be></ry></nn></nn></on></pre>	
891 REM 900 FOR VOL-15 TO 0 STEP -0.2:FOR P0=0 T 0 5:SOUND V0,P0,2,VOL:SOUND V1,P0+1,2,VO L:NEXT P0 910 FOR P1=VOL*10 TO VOL STEP -10:SOUND V0,P1,10,VOL:SOUND V1,P1+VOL,10,VOL:NEXT P1:NEXT VOL 920 RETURN 989 REM 990 REM *** DOOR BELL *** 991 REM 1000 FOR VOL=15 TO 0 STEP -0.5:SOUND V0, 29,10,VOL:SOUND V1,30,10,VOL:NEXT VOL 1010 FOR VOL=15 TO 0 STEP -0.5:SOUND V0, 35,10,VOL:SOUND V1,36,10,VOL:NEXT VOL 1020 RETURN 1089 REM 1090 REM *** BUZZER *** 1091 REM 1100 SOUND 0,40,6,10 1110 FOR DELAY-1 TO 400 1110 FOR DELAY-1 TO 400 1120 NEXT DELAY 1130 SOUND 0,0,0,0.2:RETURN	<pre></pre> <pre><uf> <uf> <uf> </uf></uf></uf></pre> <pre><uf> </uf></pre> <pre><pre><uf> <pre><pre><pre><pre><pre><pre><pre><pre></pre></pre></pre></pre></pre></pre></pre></pre></uf></pre></pre>	2150 ? ""<6>_TELEPHONE_RINGING" 2160 ? ""<7>_WHISTLING_BOMB" 2170 ? ""<8>_SPACE_SHIP" 2180 ? ""<9>_SPACE_ECHO" 2190 ? "<10>_DOOR_BELL" 2200 ? "<11>_BUZZER" 2210 ? "<12>_GALAXY_DANTINA_MUSIC" 2228 ? "<13>_THUNDER" 2230 ? "<14>_OUD_AIRPLANE" 2240 ? "<15>_PLANE_CRASHING" 2240 ? "<15>_PLANE_CRASHING" 2500 ? "<16>_RUN_LAST_PROGRAM" 3000 POKE 53279,0:? :? ""WELCHE_NR." ";:TRAP 9000:INPUT CHOICE:TRAP 40000 3010 CHOICE=INT(CHOICE):IF CHOICE<1 OR CHOICE>16 THEN RUN "D:SOUND.7" 3030 GRAPHICS 8:SETCOLOR 2,CHOICE,0:? :? :? :? :LIST CHOICE*100-10,CHOICE*100+80 3040 GOSUB CHOICE*100:? :? "DURECK E_EINE_TASTE_FUER_MENUE";:POKE 764,255 3050 IF PEEK (764)=255 THEN 3050 9000 RUN 9010 REM CASSETTEN BENUETZER MUESSEN IN	<pre><on> <nn> <nn> <ry> <be> <be> <bd> <um> <nm> <rg> <um <nm=""> <nm> <iun <rg=""> <iun <inn="" <inn<="" <iun="" td=""></iun></iun></nm></um></rg></nm></um></bd></be></be></ry></nn></nn></on></pre>	
891 REM 900 FOR VOL-15 TO 0 STEP -0.2:FOR P0=0 T 0 5:SOUND V0,P0,2,VOL:SOUND V1,P0+1,2,VO L:NEXT P0 910 FOR P1=VOL*10 TO VOL STEP -10:SOUND V0,P1,10,VOL:SOUND V1,P1+VOL,10,VOL:NEXT P1:NEXT VOL 920 RETURN 989 REM 990 REM *** DOOR BELL *** 1000 FOR VOL=15 TO 0 STEP -0.5:SOUND V0, 29,10,VOL:SOUND V1,30,10,VOL:NEXT VOL 1010 FOR VOL=15 TO 0 STEP -0.5:SOUND V0, 35,10,VOL:SOUND V1,36,10,VOL:NEXT VOL 1020 RETURN 1089 REM 1090 REM *** BUZZER *** 1091 REM 1100 SOUND 0,40,6,10 1110 FOR DELAY-1 TO 400 1120 NEXT DELAY 1130 SOUND 0,0,0,0:RETURN 1189 REM	<pre><vn> <bk>  <uf>  <uf>  <uf>  <amale< td=""><td>2150 ? "".&lt;6&gt;.TELEPHONE_RINGING" 2160 ? "".&lt;7&gt;.WHISTLING_BOMB" 2170 ? "".&lt;9&gt;.SPACE_SHIP" 2180 ? "".&lt;9&gt;.SPACE_ECHO" 2190 ? "&lt;10&gt;.DOUR_BELL" 2200 ? "&lt;11&gt;.BUZZER" 2210 ? "&lt;12&gt;.GALAXY_DANTINA_MUSIC" 2228 ? "&lt;13&gt;.THUNDER" 2230 ? "&lt;14&gt;.DUD_AIRPLANE" 2240 ? "&lt;15&gt;.PLANE_CRASHING" 2240 ? "&lt;15&gt;.PLANE_CRASHING" 2500 ? "&lt;16&gt;.RUN_LAST_PROGRAM" 3000 POKE 53279,0:? :? ""</td><td><pre><on> <nn> <nn> <nn> <ry> <be> <um> <um> <nw> <iun <inn="" <inn<="" td=""></iun></nw></um></um></be></ry></nn></nn></nn></on></pre></td></amale<></uf></uf></uf></bk></vn></pre>	2150 ? "".<6>.TELEPHONE_RINGING" 2160 ? "".<7>.WHISTLING_BOMB" 2170 ? "".<9>.SPACE_SHIP" 2180 ? "".<9>.SPACE_ECHO" 2190 ? "<10>.DOUR_BELL" 2200 ? "<11>.BUZZER" 2210 ? "<12>.GALAXY_DANTINA_MUSIC" 2228 ? "<13>.THUNDER" 2230 ? "<14>.DUD_AIRPLANE" 2240 ? "<15>.PLANE_CRASHING" 2240 ? "<15>.PLANE_CRASHING" 2500 ? "<16>.RUN_LAST_PROGRAM" 3000 POKE 53279,0:? :? ""	<pre><on> <nn> <nn> <nn> <ry> <be> <um> <um> <nw> <iun <inn="" <inn<="" td=""></iun></nw></um></um></be></ry></nn></nn></nn></on></pre>	
891 REM 900 FOR VOL-15 TO 0 STEP -0.2:FOR P0=0 T 0 5:SOUND V0,P0,2,VOL:SOUND V1,P0+1,2,VO L:NEXT P0 910 FOR P1=VOL*10 TO VOL STEP -10:SOUND V0,P1,10,VOL:SOUND V1,P1+VOL,10,VOL:NEXT P1:NEXT VOL 920 RETURN 989 REM 990 REM *** DOOR BELL *** 991 REM 1000 FOR VOL=15 TO 0 STEP -0.5:SOUND V0, 29,10,VOL:SOUND V1,30,10,VOL:NEXT VOL 1010 FOR VOL=15 TO 0 STEP -0.5:SOUND V0, 35,10,VOL:SOUND V1,36,10,VOL:NEXT VOL 1020 RETURN 1089 REM 1090 REM *** BUZZER *** 1091 REM 1100 SOUND 0,40,6,10 1110 FOR DELAY-1 TO 400 1110 FOR DELAY-1 TO 400 1120 NEXT DELAY 1130 SOUND 0,0,0,0.2:RETURN	<pre><vn> <bk>  <uf>  <uf>  <uf> </uf></uf></uf></bk></vn></pre> <pre><uf> <uf> </uf></uf></pre> <pre><uf> <uf> </uf></uf></pre> <pre><uf> </uf></pre> <pre><uf> </uf></pre> <pre><uf< pre=""> <pre><uf< p=""> <pre> <pre><uf< p=""> <pre>&lt;</pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre></uf<></pre>	2150 ? "".<6>.TELEPHONE_RINGING" 2160 ? "".<7>.WHISTLING_BOMB" 2170 ? "".<9>.SPACE_SHIP" 2180 ? "".<9>.SPACE_ECHO" 2190 ? "<10>.DOUR_BELL" 2200 ? "<11>.BUZZER" 2210 ? "<12>.GALAXY_DANTINA_MUSIC" 2228 ? "<13>.THUNDER" 2230 ? "<14>.DUD_AIRPLANE" 2240 ? "<15>.PLANE_CRASHING" 2240 ? "<15>.PLANE_CRASHING" 2500 ? "<16>.RUN_LAST_PROGRAM" 3000 POKE 53279,0:? :? ""	<pre><on> <nn> <nn> <ry> <be> <be> <bd> <um> <rg> <um> <rg> <um> <iu> <rg> <iu> <iu> <iu> <iu> <iii> <iii <ii="" <iii="" i<="" ii="" td=""></iii></iii></iu></iu></iu></iu></rg></iu></um></rg></um></rg></um></bd></be></be></ry></nn></nn></on></pre>	

,		
	Ø REM SOUND.7	
	c) 1985 by Peter Gerstner 26/11/85	<0B>
ı	1 REM	< <b>U</b> @>
	2 REM	<ur></ur>
ı	3 REM	<us></us>
ı	100 6070 10000	<jc></jc>
ı	500 POSITION 1.0: FOR ME=0 TO 8: POSITION	
ı	6, ME+21? #6; 53760+ME; "="; N (ME); "": NEXT	(DT)
ı	600 ? #6; "waehle.speicherzelle";	<dz></dz>
I	700 PRETTING E 2.2 AA/ECC CT ANV-2	<sf></sf>
I	700 POSITION 5,2:? #6; *(ESC CTL +)*:Y-2 720 S=STICK(0):IF S=14 AND Y>2 THEN POSI	\L5/
I	TION 5,Y:? #6;".":Y=Y-1:POSITION 5,Y:? #	
ı	6; CHR\$ (30): GOSUB 3000	<xn></xn>
I	730 IF PEEK (764) <>255 THEN 12000	<ch></ch>
ı	740 IF S=13 AND Y<10 THEN POSITION 5,Y:?	
ı	#6; "_": Y=Y+1: POSITION 5,Y:? #6; CHR\$ (38)	
	: 60SUB 3000	<rf></rf>
	760 IF STRIG(0)=1 THEN 720	<ag></ag>
I	800 GC=Y-2	<ub></ub>
	900 POSITION 0,11:? #6; "_trigger_auslass	
	ena.";	⟨EE⟩
ı	1000 IF STRIG(0) -0 THEN 1000	<yl></yl>
ı	1010 POSITION 5,Y:? #6;""	<se></se>
ı	1020 P=53760+6C:L\$=STR\$(P):FOR ME=1 TO 5	
ł	:IT=ASC(L*(ME,ME)):IT=IT+128:L*(ME,ME)=C HR*(IT):NEXT ME	<10>
ı	1030 POSITION 6,Y:? #6;L\$	<bp></bp>
	1040 POSITION 0,11:7 #6; "assaenderspoke	NDF /
	AAA	<nn></nn>
	2000 S=STICK(0): IF S=14 AND N(GC)<255 TH	41-12
ı	EN N(GC)=N(GC)+1	<pw></pw>
ı	2020 IF PEEK(764)<>255 THEN 12000	<ue></ue>
	2100 IF S=13 AND N(GC) >0 THEN N(GC)=N(GC	
	>-1	<qe></qe>
ı	2200 POKE P,N(SC):POSITION 12,Y:7 #6;N(6	
ı	C);""	<uu></uu>
	2300 IF STRIG(0)=1 THEN 2006	<a3></a3>
	2400 POKE 65,0:60TD 500	<qj></qj>
	3000 FOR ME=1 TO 10:POKE 53279,0:POKE 53 279,8:NEXT ME:RETURN	<lh></lh>
	10000 GRAPHICS 18:DIM N(8),L*(5):SOUND 0	\Lnz
	.0.0.0:POKE 710.154:POKE 789.284	<jm></jm>
ı	10010 REM * LITERALS IN LINE 10100 ARE I	COLLY
	NVERSE VIDEO	<vc></vc>
	10100 ? #6; "SOUND_EDITOR":? #6; "	
-	**************************************	<sz></sz>
I	11000 FOR ME=0 TO 8:N(ME)=0:NEXT ME:CLOS	
	E #1:OPEN #1,4,0,"K:":60TD 500	<us></us>
ı	12000 FOR A=0 TO 3:SOUND A,0,0,0:NEXT A:	
1	POKE 45,3	<86>
ı	12010 RUN "D:SOUND.8"	<w03></w03>
I	12020 REM	<nh></nh>
I	12030 REM Cassetten Benutzer Zeile 12010	4573
I	aendern in 'GRAPHICS G: END'	<fz></fz>
I	Listing 6. Sound-Editor. Bitte mit dem Namen	
	»D:SOUND.7« auf Diskette speichern.	
•		

@ REM SOUND.8	< NR>
I REM	<00>
2 REM	<ur></ur>
3 GRAPHICS 1+16	<ij></ij>
4 SETCOLOR 0,4,6:SETCOLOR 1,0,12:SETCOLO	
R 2,11,4:GOSUB 2000	<x5></x5>
5 FOR I=0 TO 3:SOUND 0,100,10,0:NEXT I	<xl></xl>
18 DIM B(3)	<bh></bh>
20 B(1)=128:B(2)=0:B(3)=128	<zd></zd>
48 POKE 53768,4:90=138:POKE 53763,138	<kp></kp>
50 BC=3: RESTORE 1000	<tj></tj>
100 READ T: IF T=-1 THEN RESTORE 1108: REA	
υт	<uk></uk>
105 IF PEEK (53279) = 6 THEN BRAPHICS 0:NEW	
	<bk></bk>
110 BC=BC+1:IF BC=4 THEN BC=1	<kr></kr>
115 SD=SD-10: IF SD=0 THEN SD=120	<xc></xc>
128 C=1536:POKE 53762,SD:POKE 53767,168:	
TA=1536	<@D>
127 IF B(BC)<>0 THEN C=53767:POKE 53766,	
B(BC)	<01>
138 IF T<>0 THEN POKE 53760, T: POKE 53764	
,T+1:TA=53761	<wy></wy>
140 FOR I=14 TO 2 STEP -2:POKE C,192+1:P	

OKE TA, 160+1:POKE TA+4, 160+1:NEXT, I	<10>
150 60TO 100	<lq></lq>
1996 DATA_0,0,0,0,0,0,0,0,0,0,0	<bl></bl>
1010 DATA_0,0,0,0,0,0,0,0,0,0,0	<bo></bo>
1100 DATA_162,81,81,81,81,81,91,81,81,81	
.81.81	<dr></dr>
1110 DATA_102,81,81,81,81,108,81,81,81,8	1.2.11
1,81,81	<ke></ke>
1120 DAYA_121,81,81,81,81,81,121,91,136,	
102,144,108	<cc></cc>
1138 DATA_162,121,144,108,136,192,144,10	
8,162,121,182,136	<٧@>
1140 DATA_162,121,144,100,136,102,144,10	
8,162,121,182,136	<vt></vt>
1150 DATA_204,144,217,162,243,204,243,18	
2,217,162,204,217	<et></et>
1152 DATA_162,121,144,108,136,102,144,10	******
8,162,121,182,136	<we></we>
1154 DATA-162,121,144,108,136,102,144,10	111-11
8,162,121,182,136	<wh></wh>
1160 DATA_188,9,108,108,9,121,0,0,0,8,6,	
	(WS)
1170 DATA_136,9,136,136,0,144,0,0,0,0,0,0	***
0	<ec></ec>
1180 DATA_162,0,162,0,162,0,144,0,144,0,	
144,0	<mz></mz>
1190 DATA_136,0,136,0,144,0,162,0,0,0,0,	
	<fi></fi>
1200 DATA_108,0,108,108,0,121,0,0,0,0,0,0,	
8	<wc></wc>
1210 DATA_136,0,136,136,0,144,0,0,0,0,0,0,	
0	<mc></mc>
1220 DATA_162,0,162,0,0,162,144,0,144,0,	
0,144	<03>
1230 DATA_136,0,136,144,0,162,0,0,0,0,0,0	
9	<cs></cs>
1240 DATA1	(EU)
2000 POSITION 0,3:? #6; "AVIELENADANKAFUE	
R'S"	<aa></aa>
2010 ? #6:? #6; "_MITMACHEN_UND_VIEL"	<cr></cr>
2020 ? #6:? #6: "ERFOLG_IM_MUSIK"	<ie></ie>
2030 ? #6:? #6; "PROGRAMMIEREN"	<hp></hp>
2040 ? #6:? #6: "WUENSCHT"	<wy></wy>
2050 ? #6:? #6; "DIR"	<de></de>
2060 ? #6:? #6:? #6; "peter_gerstner"	<cy></cy>
2076 POSITION 0,21:? #6; "START.=_END	
E"	<ah></ah>
2080 RETURN	<ps></ps>
Listing 7. »Oxygene« von Jean-Michel Jarre. Bitte m	it dem
Namen »D:SOUND.8« auf Diskette speichern.	it deili
Manien -D.SOUMD.OF BUI DISKette Sperchern,	

Ø REM (c) 1985 by Kemal Ezcan	<ry></ry>
1 REM	<uq></uq>
2 REM	<ur></ur>
3 REM	<us></us>
10 GOSUB 30000	<mt></mt>
15 POKE 53768,1	<pa></pa>
30 SOUND 0,0,10,10:SOUND 1,0,10,10:SOUND	
2,0,10,10:SOUND 3,0,10,10	<0V>
50 C=C+1: IF C>8 THEN C=1: READ BD,RD,SD	<80>
66 ST=ASC(S\$(C+SD,C+SD))	<xl></xl>
70 BT=ASC(B\$(C+BD,C+BD)):SOUND 3,BT,12,8	<11>
75 IF PEEK (53279)=6 THEN GRAPHICS 8: END	<xh></xh>
80 RDIS=1:A=A5C(R\$(C+RD,C+RD)):IF A<>0 T	
HEN RT=A:RL=12	<hi></hi>
98 READ T.E: IF T=-1 THEN 200	<bn></bn>
95 SOUND 0,T,10,12	<sf></sf>
100 FOR I=15 TO 4 STEP -1.6:SOUND 1,ST,1	
2, I: SOUND 2, RT, 8, RL: RL=RL-0.9*(RL)=0.6):	
NEXT I	<sa></sa>
101 SOUND 2,0,3,10	<6A>
105 IF E=0 THEN SOUND 0,0,8,0	<hm></hm>
110 GOTG 50	<q\$></q\$>
200 FOR I=15 TO 4 STEP -1:50UND 0,71,10,	
I:SOUND 1,121,10, I:SOUND 2,72,10, I:SOUND	
3,0,0,0:NEXT 1	<vt></vt>
202 FOR I=15 TO 4 STEP -1:SOUND 0,91,10,	
I:SOUND 1,121,10,I:SOUND 2,72,10,I:SOUND	
3,0,0,0:NEXT I	<vz></vz>
204 FOR I=15 TO 4 STEP -1:SOUND 0,96,10,	
Listing 8. »Digi Loo-Digi Ley« - Top-Musik auf dem A	tari

		<u></u>	
I:SOUND 1,121,10,1:SOUND 2,81,10,I:SOUND 3,0,0,0:NEXT I	<dc></dc>	1380 DATA.56,0,8,22,0,22,0,22,0,22,0,23,	<kj></kj>
206 FOR I=15 TO 0 STEP -0.8: SOUND 0,71,1		1390 DATA 56,0,16,60,0,60,0,60,0,72,1,72	
0,I:SOUND 1,121,10,I:SOUND 2,72,10,I:SOUND 3,0,0,0:NEXT 1	<am></am>	.0,81,0,91,0,81,1 1400 DATA_56,0,24,81,1,81,1,81,1,81,0,68	<0N>
210 GRAPHICS D: END	<hy></hy>	,1,68,1,68,1,68,0	<10>
220 STOP 1000 DATA.36,0,56,0,1,0,1,0,1,0,1,0,1,0,	<e0></e0>	1410 DATA.56,0,16,60,0,60,0,60,0,72,1,72,0,81,0,91,0,60,1	<hk></hk>
1,0,1,0,1	(FQ>	1420 DATA_56,0,32,60,1,60,1,60,1,60,1,60	48045
1010 DATA_56,0,56,72,0,72,0,81,0,72,0,0, 1,0,1,0,1,0,1	<wc></wc>	,0,45,0,47,0,60,1 1430 DATA,56,8,40,60,1,60,1,60,1,60,1,60	(CA>
1020 DATA_56,0,56,0,1,0,1,0,1,0,1,0,1,0,		,1,60,0,72,0,68,0	<nu></nu>
1,0,1,0,1	<fw></fw>	1440 REM REFRAIN	<w>&gt;</w>
1030 DATA_0,0,56,72,0,72,0,81,0,72,0,0,1	<yk></yk>	1450 DATA.16,0,16,60,1,60,0,60,0,72,0,53	<e0></e0>
1040 DATA_0,8,56,0,1,0,1,0,1,0,1,0,1,0,1	*****	1460 DATA.24,0,24,68,0,72,0,68,0,60,1,60	
,0,1,0,1 1045 REM VORSPEL	<uu> <pr></pr></uu>	,1,60,0,72,0,68,0 1470 DATA_0,0,0,60,0,60,0,60,0,60,0,60,0	<qy>&gt;</qy>
1050 DATA_56,0,0,72,0,72,0,72,0,81,1,81,		,91,1,91,0,53,1	<wl></wl>
0,72,1,72,0,60,0 1060 DATA_56,0,0,33,0,33,0,35,0,45,1,45,	<hu></hu>	1480 DATA_32,0,32,53,1,53,1,53,1,53,1,53 ,1,53,1,53,1,53,	<mp></mp>
0,40,1,40,0,0,1	<bp></bp>	1490 DATA_8,0,8,53,0,53,0,53,0,53,0,47,1	STIE Z
1070 DATA_56,0,0,60,0,60,0,60,0,60,1,60,	CDUIN	,47,0,47,1,47,6	<205>
0,91,1,91,0,45,0 1080 DATA_56,0,8,22,0,22,0,22,0,23,	<bh></bh>	1500 DATA,0,0,0,45,0,60,1,60,0,60,1,60,1 ,60,1,60,0,72,0	<nu></nu>
1,23,0,26,1,26,0	<kd></kd>	1510 DATA_48,0,48,68,0,72,0,68,1,68,1,68	
1090 DATA,56,9,16,60,0,60,0,60,0,72,1,72	<0H>	,0,53,1,53,0,53,1 1515 REM (CTL Q)(CTL R)(CTL R)(CTL R)1(C	<tw></tw>
1100 DATA_56,0,24,81,1,81,1,81,1,81,0,68		TL RECOTL RECOTL RESCOTL VE	<xj></xj>
1110 DATA_56,0,16,60,0,60,0,60,0,72,1,72	<w>&gt;</w>	1520 DATA_40,6,40,53,1,53,0,60,1,68,1,68	<nb></nb>
,0,81,0,91,0,60,1	<he></he>	1530 REM REFRAIN	<yv></yv>
1120 DATA_56,0,32,60,1,60,1,60,1,60,1,60	<c8></c8>	1540 DATA_16,0,16,60,1,60,0,60,0,72,0,53	/PNI
1130 DATA 56,8,40,60,1,60,1,60,1,60,1,60	1007	,1,53,0,53,0,72,0 1550 DATA,24,0,24,68,0,72,0,68,0,60,1,60	⟨EN⟩
,1,60,0,72,0,68,0	<nd></nd>	,1,60,0,72,0,68,0	<ye></ye>
1140 REM REFRAIN 1150 DATA_16,8,16,40,1,60,0,60,0,72,0,53	<yq></yq>	1560 DATA_0,0,0,60,0,60,0,60,0,60,0,60,0 ,91,1,91,0,53,1	<wk></wk>
,1,53,0,53,0,72,0	<ei></ei>	1570 DATA-32,0,32,53,1,53,1,53,1,53,1,53	
1160 DATA_24,0,24,68,0,72,0,68,0,60,1,60 ,1,60,0,72,0,68,0	<xx></xx>	,1,53,1,53,1,53,0 1580 DATA_B,0,8,53,0,53,0,53,0,53,0,47,1	<md></md>
1170 DATA_D,0,0,60,0,60,0,60,0,60,0	2007	,47,0,47,1,47,0	<dr></dr>
,91,1,91,0,53,1	<wf></wf>	1590 DATA_0,0,0,45,0,60,1,60,0,60,1,60,1	<0V>
1180 DATA_32,0,32,53,1,53,1,53,1,53,1,53 ,1,53,1,53,1,53,	<mj></mj>	,60,1,60,0,72,0 1600 DATA.48,0,48,68,0,72,0,68,1,68,1,68	/04>
1170 DATA_8,0,8,53,0,53,0,53,0,53,0,47,1		,0,53,1,53,0,53,1	<tv></tv>
,47,0,47,1,47,0 1200 DATA_0,0,0,45,0,60,1,60,0,60,1,60,1	<dm></dm>	1610 REM (CTL Q) (CTL R) (CTL R) (CTL R) 2(CTL R) 2(CTL R)	<5V>
,60,1,68,0,72,0	<n0></n0>	1620 DATA_40,0,40,53,1,53,0,60,1,60,1,60	
1210 DATA_48,0,48,68,0,72,0,68,1,68,1,68	<tq></tq>	,0,45,0,47,0,60,1 1630 DATA_40,0,40,60,1,60,1,60,1,60,1,60	<bs></bs>
1215 REM (CTL Q) (CTL R) (CTL R) (CTL R)1(C		,1,60,1,60,1,60,0	<np></np>
TL R) (CTL R) (CTL R): (CTL V) 1220 DATA-40,8,40,53,1,53,0,60,1,60,1,60	<xd></xd>	1650 DATA_0,0,0,-1,0 30000 REM INIT	<fb></fb>
,1,60,0,72,0,68,0	<mv></mv>	30010 DIM S\$(64),B\$(64),R\$(16)	<zw></zw>
1230 REM REFRAIN	<yp></yp>	30020 R#="((CTL ,)(CTL F)(CTL ,)((CTL ,) (CTL F)(CTL ,)(CTL F)(CTL F)(CTL Q)(CTL	
1240 DATA_16,0,16,60,1,60,0,60,0,72,0,53 ,1,53,0,53,0,72,0	⟨EH⟩	©) ( ( ( "	<wi></wi>
1250 DATA_24,0,24,68,0,72,0,68,0,68,1,60		30030 BS="(CTL R)(CTL R)(CTL R)(CTL R)(CTL R)(CTL R)	
,1,60,0,72,0,68,0 1260 DATA.0,0,0,60,0,60,0,60,0,60,0,60,0	<xw></xw>	FOTE RECTE RECTE REGOOD COTE , COTE , E	
,91,1,91,0,53,1	<me></me>	(CTL ,) (CTL ,) ?????nnnnnnnn??????? (CTL ,	
1270 DATA_32,0,32,53,1,53,1,53,1,53,1,53 ,1,53,1,53,1,53,	<mi></mi>	TOTAL TOTAL TOTAL NO COTE NO C	
1280 DATA_8,0,8,53,0,53,0,53,0,53,0,47,1		TE NECTE NECTE NECTE NE	<vo></vo>
147,8,47,1,47,8 1298 DATA_0,0,0,45,0,60,1,60,0,60,1,60,1	<dl></dl>	30040 S\$="a0a0a0a0H\$H\$(CTL_B)?(CTL_B)?a0 a0s:s:W+W+(CTL_B)?(CTL_B)?H\$H\$H\$H\$(CTL_B	
,60,1,60,0,72,0	<0P>	>2(CTL B)2(CTL B)2(CTL B)2W+W+W+W+CTL N	
1300 DATA_48,0,48,68,0,72,0,68,1,68,1,68	<tp></tp>	>{CTL N>{CTL N>{	(KV)
.0,53,1,53,0,53,1 1310 REM (CTL Q)(CTL R)(CTL R)(CTL R)2(C	1167	30050 GRAPHICS 3+16:SETCOLOR 4,2,6	<ny></ny>
TL R3 (CTL R) (CTL R)	<sp></sp>	30060 GRAPHICS 18: SETCOLOR 4,0,0: SETCOLO	<gw></gw>
1320 DATA_40,0,40,53,1,53,0,60,1,60,1,60,1,60,0,45,0,47,0,60,1	<bm></bm>	R 0,0,15 30070 ? #6:? #6;"M_C_S_C#######":? #	TOR/
1330 DATA_40,0,40,60,1,60,1,60,1,60,1,60		6;"	(1.11)
,1,60,1,60,1,60,0 1340 REM WIEDERHOLUNG	< LM>	WAREAM###*am#" 30075 ? #6;"AAAAAAAAA#AA####";? #6;"AP	<lv></lv>
1350 DATA_56,0,0,72,0,72,0,72,0,81,1,81,		RESENTANA**********************************	<kx></kx>
0,72,1,72,0,60,0 1360 DATA_56,0,0,33,0,33,0,35,0,45,1,45,	<ia></ia>	30080 POSITION 0,7:0 #4; "DIGIALOOA/ADIGI	<wy></wy>
0,40,1,40,0,0,1	<8V>	30085 ? #6; "_(C)_BY_KEMAL_EZCAN"	<et></et>
1370 DATA_56,0,0,60,0,60,0,60,0,60,1,60,	ZBNIS	30090 C=8: RETURN Listing 8. »Digi Loo-Digi Ley« (Schluß)	<vq></vq>
0,91,1,91,0,45,0	<bn>  </bn>	mounty or any mounty treat (occurre)	

## Aktion mit Action

Action zählt zu den schnellsten Programmiersprachen für Atari-Computer. Wer auf hohe Geschwindigkeiten Wert legt, aber nicht in Maschinensprache programmieren will, sollte Action in Erwägung ziehen.

n letzter Zeit machte eine Programmiersprache von sich reden, die ursprünglich als Sprache für Systementwickler konzipiert war: »C«. Sie wurde Anfang der siebziger Jahre von Dennis Ritchie aus BCPL weiterentwickeit. Als Standardwerk zur Programmierung in »C« wird daher auch oft das Buch von Kernighan/Ritchie (»The C Programming Language«) genannt. Ritchie war später auch an der Entstehung von UNIX beteiligt.

Es gab bereits mehrere Versuche, die Sprache C auf den Atari anzupassen: Das »Deep Blue C«-System ist weitgehend an einen klassischen C-Compiler angelehnt, Allerdings sind hier nicht nur beim Befehlsumfang, sondern auch bei der Arbeitsgeschwindigkeit Abstriche zu machen. Ein anderer Weg wurde mit C/65 begangen, Bei diesem System wird der Quelltext in einen Assemblertext compiliert, der dann mit MAC/65 (siehe Beitrag in diesem Sonderheft) weiter bearbeitet werden muß.

Unser Thema ist aber die Programmiersprache »Action«, die von OSS stammt. Diese Sprache enthält zwar viele Elemente von C, ist aber dennoch »anders« genug, um einen eigenen Namen verdient zu haben.

Action wird (womit wir schon beim ersten Unterschied zu konventionellen C-Compilern sind) auf einem Programmmodul geliefert. Wie bei allen anderen Cartridges von OSS handelt es sich dabei um ein 16 KByte-Super-Cartridge. Es belegt aber trotzdem nur 8 KByte-RAM-Speicher. Bei Verwendung von DOS XL kann der freie Speicherplatz sogar noch weiter erhöht werden. Der nächste Vorteil ist (genau wie bei MAC/65), daß sich Editor und Compiler gleichzeitig im Speicher befinden. Von der Konzeption des Systems her läßt sich Action also mit

Turbo-Pascal vergleichen, bei dem ebenfalls Editor, Compiler und Quelltext gleichzeitig im Speicher vorliegen. Ein separater Linker ist nicht nötig.

Da Action, genau wie C, nicht mit Zeilennummern arbeitet, hat man dem Action-System einen stark an Textprogramme angelehnten Editor eingebaut.

Grundsätzlich stehen alle normalen Fähigkeiten des Bildschirmeditors zur Verfügung. Allerdings kann jede Zeile maximal 240 Zeichen lang sein, so daß man bei Schleifen und Kommentaren genügend Platz zum Einrücken hat. Da die Darstellung auf dem Bildschirm auf nur 40 Zeichen pro Zeile begrenzt ist (Bild 1), ist natürlich immer nur ein Teil des Listings zu sehen. Bei den Zeilen, die über den Rand des Bildschirms hinausreichen, ist jeweils das letzte Zeichen invertiert. Bewegt man nun den Cursor über den rechten Rand des Bildschirms, wird die Zeile, in der man sich befindet, nach links gescrollt (Bild 2). Weitere Fähigkeiten des Editors sind Funktionen zum Suchen und Ersetzen von Begriffen, der Einsatz eines zweiten Bildschirmfensters, in dem man eine andere Datei bearbeiten kann, und ähnliches mehr.

Das Fehlen von Zeilennummern beim Programmieren erschwert es oft. bestimmte Programmteile wiederzufinden: Der Action-Editor erlaubt aber, an beliebigen Stellen Markierungen (Tags) zu setzen, die man dann später mit nur zwei Tasten wieder anwählen kann.

Sämtliche Parameter wie Zeilenlänge und Fenstergröße etc. sind veränder-

Den Mittelpunkt des Action-Systems bildet der Monitor, von dem aus man Editor, Compiler und DOS aufrufen kann, Außerdem können einzelne Speicherzellen verändert und ganze Speicherbereiche gelistet werden. Wichtig beim Testen eines Programms ist der TRACE-Modus. Während des Programmablaufs erlaubt er die Ausgabe von Namen und Parametern ieder aufgerufenen Prozedur auf dem Bildschirm.

Die Programmiersprache Action ist weitgehend an C angelehnt. So haben beide grundsätzlich ähnliche Datentypen und eine ähnliche Syntax.

Da zwischen den einzelnen Befehlen nur Leerzeichen stehen brauchen, ist man in der Aufteilung des Quelltextes völlig frei.

Natürlich ist es empfehlenswert, bei Verschachtelungen, wie bei anderen Programmiersprachen auch, dazwischenliegende Zeilen einzurücken. So werden Programme übersichtlich und überschaubar.

An Datentypen sind in Action zunächst einmal BYTE und CHAR (ein-Byte), INT (2-Byte-Integer zwischen -32768 und 32767) und CARD (desaleichen zwischen 0 und 65535) vertreten, Bei der Deklaration von Variablen kann man (muß aber nicht) angeben, welche Speicherstelle dazu benutzt werden soll. Beispiel: »BYTE chbase=\$02F4«

Mit dieser Zeile würde-man eine 1-Byte-Variable erzeugen, die genau auf der Adresse \$02F4 liegt. Bei einer Wertzuweisung zu chbase wird die Adresse \$02F4 geändert. Das Kommando »chbase=\$CC« schaltet also den internationalen Zeichensatz ein. \$02F4 fungiert nämlich als Basisregister für den Zeichensatz und \$CC als High-Byte der Anfangsadresse des Zeichensatzes. Wo Basic umständliche POKE-Befehle erfordert, kann man hier sehr viel eleganter programmieren. Gleiches gilt natürlich für die anderen

Neben den fundamentalen Datentypen gibt es in Action auch erweiterte Arten von Variablentypen. Fangen wir mit dem Typ POINTER (Zeiger) an. Davon abgesehen, daß die Deklaration ein wenig anders abläuft (»CHAR POIN-TER pnt« statt »CHAR \*pnt«), ist die Anwendung der Pointer in Action genauso wie in C gelöst. Mit »pnt=@var« erhält man die Adresse einer Variablen (in C:»pnt=&var«), mit »pnt a erhält man die Adresse, auf die pnt zeigt (in C müßte der Befehl dann \*pnt< lauten).</p>

Ein weiterer wichtiger Datentyp ist das Feld (Array). Im Gegensatz zu C verwendet man in Action allerdings nur eindimensionale Arrays. Sie werden außerdem nur aus den fundamentalen Datentypen, also CHAR, BYTE, INT und CARD, aufgebaut. Allerdings kann man mit einem Trick auch Arrays aus den erweiterten Datentypen verwenden.

In Action lassen sich auch eigene Datentypen definieren; sie heißen Records. Man benutzt dazu die Anweisung TYPE, während es in C STRUCT heißt. Der Vorteil der Records liegt darin, verschiedene Datentypen unter einem gemeinsamen Oberbegriff zusammenfassen zu können.

Wie schon bei der Erörterung der fundamentalen Datentypen zu sehen war, erlaubt Action eine sehr maschinennahe Programmierung. Alle numerischen Konstanten können auch in hexadezimaler Schreibweise eingegeben werden, und auch samtliche Funktionen zur Manipulation einzelner Bits sind verfügbar: »&« (logisches AND), »%« (OR), "!" (XOR), LSH (left shift) und RSH (right shift). Leider fehlt eine angenehme Eigenschaft von C. Es ist nicht möglich, Variablen an beliebigen Stellen im Programmtext durch zwei kurze Zeichen zu in- oder dekrementie-



Immerhin kann man bei Veränderungen einer Variablen den Namen auf der rechten Seite des Gleichheitszeichens weglassen, vorausgesetzt man schreibt stattdessen zwei Gleichheitszeichen (Beispiel: »Zaehler = +1« erhöht die Variable Zaehler um eins).

Ein wichtiger Gesichtspunkt bei der Beurteilung einer Sprache sind die verfügbaren Strukturen für Schleifen und Verzweigungen. Action ähnelt in dieser Hinsicht weniger C, sondern eher Basic oder Pascal. Für Verzweigungen dient natürlich das Kommando IF. Hierbei ist die Anzahl der Befehle, die THEN folgen dürfen, völlig offen. Am Ende des durch THEN eingeleiteten Blocks muß allerdings ein Flistehen. ELSE ermöglicht es dann, den entgegengesetzten Fall abzufangen. Die Worte DO und OD beginnen beziehungsweise beenden einen Schleifenblock. Diese beiden Ausdrücke ersetzen die geschweiften Klammern, die üblicherweise zwischen einfachen Anführungszeichen eingebunden sind. Dabei kann man ohne Bedingungen arbeiten und erhält dann eine unendliche Schleife - oder aber WHILE, FOR oder UNTIL benutzen, um eine Bedingung für die Schleife festzulegen. Einen vorzeitigen Abbruch von Schleifen bewirkt der Befehl EXIT.

Was eine strukturierte Programmiersprache sein will, muß natürlich über Prozeduren, Funktionen und lokale Variablen verfügen. Während sich Prozeduren und Funktionen in C nur darin unterscheiden, daß Funktionswerte zurückgegeben werden (oder nicht), werden sie in Action bereits bei der Deklaration unterschieden.

Bei Prozeduren geht das folgendermaßen vor sich: Zunächst wird mit PROC angezeigt, daß eine Prozedurdeklaration stattfindet. Es folgen der Name der Prozedur und (in Klammern) Datentypen und Namen der übergebenen Parameter. Damit sind die übergebenen Parameter automatisch als lokale Variablen für diese Prozedur definiert.

Beendet wird der Vorgang mit dem Kommando RETURN, Prozeduren werden wie Kommandos aufgerufen (Beispiel: »Wait (30)«).

Funktionen unterscheiden sich von Prozeduren darin, daß sie einen Funktionswert an das aufrufende Programm zurückliefern können. Ein Aufruf einer Funktion sieht deshalb auch ein wenig anders aus. Beispiel: »X=STICK (0)«. Funktionsdeklarationen muß zunächst der Typ des ermittelten Endresultats angegeben werden. Es folgt dann das Kommando FUNC, der Name der Funktion und die Liste der übertragenen Parameter (genau wie bei Prozeduren). Der einzige weitere Unterschied ist, daß zum Schluß dem Befehl RETURN noch ein Parameter übergeben wird, der dann als Funktionswert gilt.

Genau wie bei einem C-System werden auch zum Action-Compiler die wichtigsten Ein- und Ausgabefunktionen in Form einer Unterprogrammbibliothek mitgeliefert. Der große Unterschied ist allerdings, daß diese Routinen nicht, wie sonst üblich, auf Diskette beiliegen und dann in das compilierte Programm eingebunden werden, sondem im ROM des Action-Cartridges liegen. Dies hat zur Folge, daß compilierte Programme, wenn sie auf vordefinierte Funktionen oder Prozeduren zurückgreifen, nur mit eingestecktem Cartridge laufen.

Weiterhin gibt es, passend zum Action-Modul, eine Unterprogramm-bibliothek. Darin enthalten sind zunächst Befehle zur Ausgabe sämtlicher Datentypen (inklusive vor PrintF). Daneben findet man, auch jeweils für alle Datentypen, den INPUT-Befehl. Ebenso sind alle anderen normalen Einund Ausgabefunktionen, wie GET, PUT, OPEN, CLOSE, XIO, NOTE und POINT,

bereits als Prozedur definiert. Leider sucht man die recht wichtigen Befehle BGET und BPUT vergeblich. Grafikund Soundbefehle, die zum größten Teil ihren Basic-Vorbildern entsprechen, fehlen aber nicht.

Da Action nicht ohne weiteres in der Lage ist, Zeichenketten zu verarbeiten, sind auch hierfür einige nützliche Prozeduren vordefiniert.

Wie wir bisher gesehen haben, hat Action zwar viele Eigenschaften von C, eine vollständige Implementation liegt aber beileibe nicht vor. Andererseits wartet Action mit Erweiterungen auf, die besonders auf dem Atari von sehr großem Nutzen sind. Dazu zählt zum Beispiel auch die Festlegung der Adresse einer Variablen.

Wer zeitkritische Programmteile doch lieber in Maschinensprache schreiben will, dem stehen zweierlei Wege offen. Für kurze, verschiebbare Programmteile empfiehlt es sich, den Maschinencode direkt als Datenblock in den Quelltext einzubinden. Längere Programme kann man ab einer festen Adresse assemblieren und die zugehörige Prozedur im Quelltext einzig und allein durch Festlegung ihrer Anfangsadresse definieren.

Für den professionellen Programmierer ist es wichtig zu wissen, daß er die volle Kontrolle darüber hat, wie der Compiler den Speicherplatz aufteilt. So läßt sich dann der erzeugte Code in ein EPROM brennen.

Weiterhin ist für den ernsthaften Einsatz wichtig, die Verwendung der im ROM integrierten Unterroutinen unterbinden zu können. Man definiert einfach die Routinen selbst neu. Einfacher ist es allerdings, auf die Action-Runtime-Library zurückzugreifen. Der Anwender bekommt eine Reihe vordefinierter Dateien, die mit dem INCLUDE-Befehl in die selbstverfaßten Programme eingebunden werden.

```
; Frase8(127-x1*y0, 127*y1)

FI

RETURW

PROC GetParam(STRING param, CARD ARRAY @ CARD resultC
STRING numBuf(0)=$550

Print(param)
Print(" = ")
Print(" = ")
Print(" initial = ")
Print(" initial = ")
Print(" initial = ")
F1

F1

Print("Enter new value: ")
resultC = InputC()
If numBuf(0)10 THEM
AGINOL* (0)1033 ACS
```

Bild 1. Eine Zelle kann bis zu 240 Zeichen lang sein. Da sich nur 40 Zeichen auf dem Bildschirm darstellen lassen, wird die Zeile, in der sich der Cursor befindet, gegebenenfalls nach links oder rechts gescrollt.

```
; Erase8(127-x1+y0, 127+y1)

FI
RETURN

GSTRING param, CARD ARRAY cur, initial)

Lard resultC

SYRING numbuf(0)=$550

Print(param)
Print(" = ")
PrintC(cur^)

If initial THEM
Print(", initial = ")
PrintCE(initial^)

ELSF
Pute()

Print("Enter new value: ")
resultC = Inputc()
IF numbuf(0)10 THEM
ACTION: (C)1983 ACS
```

Bild 2. Die mit dem Cursor gekennzeichnete Zeile ist bereits nach links gescrollt. Alle anderen Zeilen befinden sich an der ursprünglichen Position. So bleibt der zu bearbeitende Text übersichtlich.

Unter dem Strich gesehen handelt es sich bei Action (etwa 250 Mark) um eine professionelle Programmiersprache mit Arbeitsgeschwindigkeiten, die auf dem Atari bisher für eine höhere Programmiersprache als unerreichbar galten: Ein Benchmarkprogramm zur Berechnung der ersten 1000 Primzahlen war in Action nur 40 Prozent langsamer als die in Maschinensprache geschriebene Version.

Ein Grund für diese enorme Geschwindigkeit ist darin zu suchen, daß der Compiler intern völlig anders vorgeht als ein normaler C-Compiler. Durch diese vereinfachte maschinensprachnähere Arbeitsweise ist leider auch die Fähigkelt verlorengegangen, rekursiv zu arbeiten.

Als weiteren Zusatz zu Action bekommt man für etwa 100 Mark die »Programmer's Aid«-Diskette. Auf dieser Diskette findet man dann schmerzlich vermißte Befehle wie CIRCLE, Kommandos für Player/Missile-Grafik oder Fließkommaverarbeitung sowie einige Demoprogramme.

Action hat also für jeden etwas zu bieten: Dem Umsteiger von Basic wird eine schnelle, komfortable Sprache geboten, die auch dem Profi-Programmierer noch voll ausreicht. Und wer später plant, auf einem »großen« Computer in C weiterzumachen, kann sich bereits mit seinem Atari einarbeiten.

(Julian F. Reschke/wb)

Bezugsquellen CompyShop, Greisenaustr 29, 4330 Mühlheim/Ruhr, Tel (0208) 497169 Münzerlicher, Tölzer Str. 4, 8150 Hozzkirchen, Tel. (08024) 1814

## Das Textverarbeitungs-Sextett

Sechs verschiedene Textverarbeitungen auf dem Prüfstand. Welche ist die beste?

en Heimcomputer als Schreibmaschine zu nutzen ist wohl eine der häufigsten ernsthaften Anwendungen im Heimbereich. Insgesamt sechs verschiedene Textverarbeitungen sollen deshalb verglichen werden, damit Sie beim Kauf die richtige Wahl treffen.

Eine der ältesten Textverarbeitungen für den Atari ist wohl der »Atari-Schreiber«. Er ist auf Modul und neuerdings auch auf Diskette lieferbar. Der »Atari-Schreiber« ist die deutsche Version des amerikanischen »Atari-Writers« und deshalb auch auf deutsche Verhältnisse abgestimmt, Sowohl das Handbuch wie auch die Benutzerführung im Programm sind in deutscher Sprache abgefaßt. Ebenso ist der Zeichensatz und sogar die Tastaturbelegung der deutschen Norm angepaßt. Wer sonst viel mit der Schreibmaschine umgeht, wird darüber erfreut sein. Alte Hacker auf dem Atari können sich allerdings anfangs wahrscheinlich vor Tippfehlern nicht mehr retten. Deshalb liegen dem Programm auch Aufkleber für die Tastatur bei, die die umbelegten Tasten entsprechend kennzeichnen.

Die Bedienungsanleitung zum »AtariSchreiber« besteht aus drei Teilen. Das eigentliche Handbuch beinhaltet eine ausführliche Beschreibung des Programms und gibt Tips zum Umgang mit dem System. Zudem liegt eine Referenzkarte bei, in der alle Befehle von »Atari-Schreiber« kurz, aber informativ aufgeführt sind. Zum schnellen Nachschlagen von bestimmten Befehlen benutzt man am besten die Referenzkarte, die sich als nützliches Hilfsmittel beim Umgang mit der Textverarbeitung erweist.

Der »Atari-Schreiber« wird von einem Hauptmenü aus gesteuert, das die wesentlichen Funktionen zur Bearbeitung eines Textes enthält (Bild 1). Dieses Menü erscheint sofort nach dem Laden des Programms und erleichtert so den ersten Einstieg in die Textverarbeitung. Auch ohne lange im Handbuch nachschlagen zu müssen, kann man gleich mit der Eingabe des ersten Textes beginnen. Man betätigt einfach die Taste »N« und gelangt in den Schreib-Modus. Am oberen Bildschirmrand ist nun eine Zeile zu sehen, welche die grundlegenden Druckparameter, wie linke und rechte Randbegrenzung, oder den Zeilenabstand enthält. Am unteren Bildrand befindet sich die Anzeige über den aktuellen Cursorstand. Leider beziehen sich diese Angaben nur auf die Bildschirmposition und nicht auf die tatsächliche Position des laufenden Textes. So ist eigentlich nur die Spaltenangabe von Nutzen. Insgesamt lassen sich 21 Zeilen von je 35 Zeichen auf einmal auf dem Bildschirm darstellen. Ein horizontales Scrollen über die 35. Spalte hinaus findet nicht statt. Die Darstellung von Tabellen kann daher zum Problem werden. Die Formatierung eines Textes auf dem Bildschirm ist nicht identisch mit dem späteren Ausdruck. Der Text wird nur dem Bildschirmformat angepaßt: von den Einstellungen, die Ränder, Blocksatz und ähnliches betreffen, muß man sich leider erst auf dem Papier überraschen lassen.

Die Druckeranpassung ist für die Atari-Drucker 1020, 1025 und 1027 sowie für den Epson FX-80 über ein kleines Untermenü vorzunehmen. Andere Drucker werden vom Atari-Schreiber nicht unterstützt, so daß man in diesem Fall entweder für entsprechende Steuerzeichen im Text sorgen muß oder sich selbst ein passendes Drucksteuerprogramm schreibt.

Beim »Atari-Schreiber« ist die Ausgabe von Serienbriefen vorgesehen. Daher kann man einen Text auch mit Variablen versehen. Braucht man beispielsweise ein Rundschreiben mit gleichlautendem Text, aber persönlicher Anrede, so wird die Anrede durch eine Variable ersetzt. Beim Ausdruck des Rundschreibens schließlich steht anstatt der Variablen der eigentliche Text. Dieser wird einfach einer Datei von der Diskette entnommen, die die erforderlichen Daten enthält.

## Stern oder Sternschnuppe?

Wer größeren Wert auf komfortable Benutzerführung legt, ist mit dem »Star-Texter« gut beraten. Allerdings meldet sich »StarTexter« mit lautem Getöse und grafischen Spielereien. So würde man eher ein neues Spiel an Stelle eines Textprogramms vermuten. Doch der erste Schein trügt. Wie schon der »Atari-Schreiber« ist auch dieses Programm auf den deutschen Benutzer zugeschnitten. Man verfügt also über einen deutschen Zeichensatz, kann aber auch beliebige andere Zeichensätze nachladen. Auf der Diskette ist deshalb ein spezielles Programm mit dem Namen »StarFont« enthalten. Damit kann man sich seine eigenen Zeichen definieren. Auf diese Art lassen sich natürlich auch spezielle grafische Symbole in einen Text aufnehmen, denn alle Sonderzeichen werden nicht nur auf dem Bildschirm dargestellt, sondern auch in dieser Form auf dem Drucker ausgegeben. Die Tastaturbelegung kann man sich aussuchen. Entweder arbeitet man mit der regulären Belegung der Tasten, oder man wandelt die Tastatur des Atari in eine deutsche Schreibmaschinentastatur nach DIN um, je nach eigenen Bedürfnissen und Gewohnheiten.



Nach dem Laden des Programms befindet man sich zuerst im normalen Textmodus. Die Darstellung des Textes erfolgt in 21 Zeilen mit je 40 Zeichen (Bild 2). Durch horizontales Scrollen können, je nach vorhergehender Randeinstellung, allerdings bis zu 80 Zeichen nebeneinander dargestellt werden. Die Spalte und Zeile, in der man sich befindet, entnimmt man der Anzeige am unteren Bildrand. Anders als beim »Atari-Schreiber« wird hier die tatsächliche Position im Text und nicht die Bildschirmposition angezeigt.

Drückt man die ESC-Taste, gelangt man in den sogenannten Control-Modus, Mit den Sondertasten START, OPTION und SELECT kann man eines der drei Menüs von »Starlexter« aufrufen. Das erste dieser Menüs bezieht sich rein auf Diskettenoperationen (Bild 3). Hier kann man beispielsweise Texte laden und speichern oder sich das Inhaltsverzeichnis einer Diskette ansehen. Daneben lassen sich auch eine Reihe von verschiedenen DOS-Operationen durchführen, wie das Löschen und Umbenennen von Files. Von hier aus lädt man auch andere Zeichensätze oder speichert die momentane Parametereinstellung auf Diskette.

Das zweite Menü ist so umfangreich, daß es gleich drei Bildschirmseiten in Anspruch nimmt. Hier werden alle Grundeinstellungen für die Textdarstellung vorgenommen, wie die Zeilenlänge oder der Blocksatz. Aber auch die farbliche Darstellung auf dem Bildschirm kann an die persönlichen Bedürfnisse angepaßt werden.

Das dritte Menü schließlich ist für den Ausdruck und das Formatieren von Texten zuständig. Die Anpassung an einen bestimmten Drucker erfolgt aber nicht hier, sondern bereits in einem eigenständigen Installationsprogramm. In diesem Basic-Programm, das unter dem Namen »INSTALL« auf Diskette abgelegt ist, wählt man entweder gängige Druckertypen direkt an, oder man definiert sich seine eigene Anpassung an den speziellen Drucker. Erwähnenswert ist hierbei, daß man nicht unbedingt über ein Centronics-Interface verfügen muß, um Texte an den Drucker zu schicken. Mit einem einfachen Kabel, dessen Herstellung im Handbuch beschrieben ist, kann man seinen Drucker auch an die Joystickports des Atari-Computers anschließen. Einzige Einschränkung des Druckbetriebes ist dabei jedoch, daß nicht alle Grafikzeichen des erweiterten Zeichensatzes ausgedruckt werden. Dies stört bei normaler Textverarbeitung aber kaum. Vor dem Drucken kann man sich den Text auch in 80-Zeichen-Darstellung betrachten, also in der Form, wie er später tatsächlich auf dem Papier erscheint,

Diese Darstellung ist zwar aufgrund der mangelnden Auflösung des Atari nur schwer lesbar und nicht zum Editieren geeignet, vermittelt aber einen guten optischen Überblick über den fertigen Text

Das Formatieren von Texten direkt auf dem Bildschirm ist mit »StarTexter« auch möglich. Jedoch ist dies bei umfangreicheren Texten eine sehr zeitraubende Angelegenheit. Hinzu kommt, daß aufgrund der Textformatierung recht verschwenderisch mit dem Speicherplatz, sowohl auf Diskette als auch im RAM-Speicher, umgegangen wird. Ein Beispiel: Begrenzt man den rechten Rand eines Textes auf Spalte 40, so wird trotzdem jede Zeile mit einer Länge von 80 Zeichen auf Diskette abgelegt, 40 Leerzeichen pro Zeile werden also umsonst gespeichert. Man benötigt folglich doppelt soviel Speicherplatz als eigentlich notwendig. Abhilfe schafft hier nur das ständige Umformatieren vor dem Speichern und nach dem Laden eines Textes, das aber, wie bereits erwähnt, recht lange dauert. Auch der Textspeicher leidet unter dieser Art der Darstellung, Maximal kann er 20000 Zeichen aufnehmen. Die Anzahl der Zeilen, die ein Text umfassen darf, ist jedoch auf 250 beschränkt. Schreibt man seinen Text mit einer Breite von 80 Spalten, so verfügt man auch tatsächlich über 80 mal 250 oder 20000 Zeichen Speicherplatz. Bei einer Randeinstellung bis zur Spalte 40 beispielsweise kann man nur mehr 40 mal 80, also 10000 Zeichen im Textspeicher unterbringen. Ideal scheint diese Lösung also nicht zu sein.

## Textprogramm aus dem Alpenland

Das nächste Textverarbeitungsprogramm für den Atari nennt sich »Austro.Text«. Wie der Name schon vermuten läßt, stammt dieses Programm aus Österreich. Auch hier verfügt man neben einer deutschsprachigen Anleitung über einen deutschen Zeichensatz. Auf eine Umbelegung der Tastatur wurde jedoch bewußt verzichtet, um. laut Handbuch, Umgewöhnungsprobleme zu vermeiden. Ein Menû sucht man bei diesem Programm vergeblich. Alle Befehle zum Bearbeiten eines Textes muß man leider im Kopf haben beziehungsweise im Handbuch nachlesen. Bei der Befehlsausführung erscheint lediglich eine Art Statuszeite, in der man die Eingaben einträgt. So ist es gerade in der Einarbeitungsphase doch recht mühsam, mit den vielen verschiedenen Befehlen von »Austro.Text« zurechtzukommen. In der Praxis sieht das so aus:

Will man beispielsweise den linken Rand des Textes beim Ausdruck auf Spalte 10 setzen, so fügt man das Kommando ».LM 10« in den Text ein. Entsprechend würde eine Begrenzung des rechten Randes auf Spalte 50 dann ».RM 50« lauten. Wie man sieht, sind diese Abkürzungen dem Englischen entlehnt (LM = Left Margin oder linker Rand), die sich – mit entsprechenden Englischkenntnissen – gut einprägen.

Bei diesem Programm wird ebenfalls kein separates Centronics-Interface zur Druckerausgabe benötigt. Mit einem speziellen Kabel, das man aber, anders als bei »StarTexter«, fertig kaufen muß, kann man einen Drucker mit Centronics-Schnittstelle auch über die Joystickports ansteuern. Zum Drucken eines Textes betätigt man gleichzeitig die START- und die P-Taste. Nun gibt man den Handlernamen (Gerätebezeichnung) ein, auf dem der Druck erfolgen soll. Der Handler »P:« spricht also die normale Centronics-Schnittstelle an. während »J:« den Ausdruck über die Joystickports veranlaßt. Auf diese Weise kann man einen Text bei entsprechender Wahl des Handlers natürlich auch als LIST-File auf Diskette oder Kassette ablegen und dann weiterverarbeiten. Spricht man den Handler »T:« an, so erscheint der Text auf dem Bildschirm im 80-Zeichen-Format. Die Darstellung entspricht dann genau dem Format, in dem der Text auf dem Drukker ausgegeben wird. Um Texte in diesem Modus darstellen zu können, muß man jedoch über mindestens 5500 Byte freien Speicherplatz verfügen. Wie auch in der normalen Textdarstellung (Bild 4) kann man übrigens hier den Feinscroll-Modus wählen, in dem der Text fließend über den Bildschirm rollt.

Zur Anpassung an die verschiedenen Druckertypen benutzt man auch hier ein eigenes Programm namens »Printer Editor«. Dieses Programm wird allerdings bereits beim Booten von »Austro. Text« in den Speicher geladen und belegt dann rund 7500 Byte Textspeicher. Das bedeutet, daß man noch 7500 Byte freien Speicher benötigt, um noch Änderungen zur Druckeranpassung vornehmen zu können. Ansonsten bleibt nur, den momentanen Text zu speichern und die Diskette neu zu booten.

## A ohne Ä

»A-Text« ist als Textverarbeitungsprogramm für den Atari auch schon etwas länger auf dem Markt. Erfreulich ist die Tatsache, daß sich inzwischen das früher kritisierte Handbuch verbessert hat. Es erscheint nach wie vor in Deutsch, ist aber wesentlich übersicht-



Blid 1. Das Hauptmenü vom »Atari-Schreiber« umfaßt alle wichtigen Funktionen.

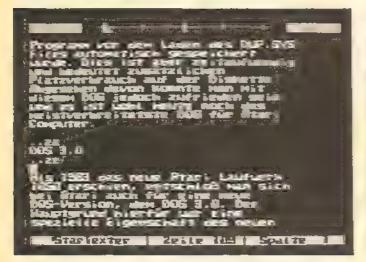


Bild 3. Die Textdarstellung von »StarTexter« mit geändertem Zeichensatz. Texte lassen sich, selbst mit anderem Zeichensatz, auf einem Drucker ausgeben.

licher geworden. Auf die Darstellung der deutschen Umlaute und Sonderzeichen auf dem Bildschirm muß man als »A-Text«-Benutzer jedoch verzichten. Statt dessen fügt man spezielle Grafikzeichen in den Text ein, die später bei der Druckerausgabe als Umlaute wiedergegeben werden.

Wie schon bei »AustroText« vermißt man bei diesem Programm die Menü-Steuerung. Die Kommandos müssen also auch hier im Kopf parat sein und dann in eine eigene Statuszeile eingegeben werden. Daß dies aber etwas langwierig werden kann, soll folgendes Beispiel aus dem Handbuch zeigen: Will man einen Text beispielsweise unterstreichen, so muß man folgende Sequenz eingeben:

»CTRL-L U CTRL-R CTRL-A CTRL-R ESC | RETURN«

Ähnlich sieht der Befehl zum Beenden des Unterstreichens aus. Beim »StarTexter« muß man vergleichsweise nur ein inverses A mit nachfolgender Null im Text einfügen, um denselben Effekt zu erzielen. Wie man sieht, ist die



Bild 2. Von diesem Menü aus lassen sich beim »Starlexter« sämtliche Diskettenoperationen ausführen.

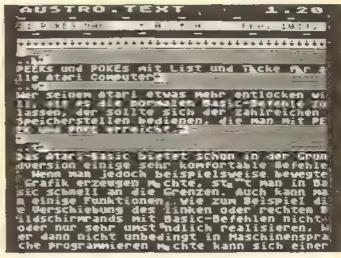


Bild 4. »Austro.Text« im Schreibmodus. Alle Leerzeichen und Leerzeilen werden mit Hilfe einer Schraffur optisch deutlich hervorgehoben.

Handhabung von »A-Text« etwas gewöhnungsbedürftig.

Die Textdarstellung auf dem Bildschirm erfolgt in 20 Zeilen mit je 38 Zeichen. Ab der 20. Spalte beginnt der Bildschirm jedoch horizontal zu scrollen, so daß Zeilen mit einer maximalen Länge von 255 Zeichen ohne Zeilenvorschub eingegeben werden können. Vor allem bei Tabellen ist dies sehr nützlich und dient der Übersichtlichkeit.

Am oberen Bildrand sind ständig einige Informationen eingeblendet (Bild 5). So erfährt man die Zahl der bisher eingegebenen Zeichen, die Spalte, in der sich der Cursor befindet, und den restlichen Speicherplatz, der für den zu bearbeitenden Text zur Verfügung steht. Zudem ist die Größe des Kopierspeichers angegeben. Dieser Speicher umfaßt rund 4000 Zeichen und dient vor allem zur Verschiebung von Textblöcken. Am unteren Bildrand befindet sich die Statuszeile, die man über die ESC-Taste erreicht. Hier gibt man alle Kommandos ein, wie beispielsweise das Laden und Speichern von Texten.

Zur Anpassung des Programms an verschiedene Drucker existiert kein spezielles Programm. Alle Steuersequenzen für den Drucker müssen daher direkt im Text eingegeben werden. Laut Handbuch soll auch »A-Text« über die Joystickports mit einem Drucker in Verbindung treten können. Wie dies genau geschehen soll, bleibt selbst nach mehrmaligem Lesen des entsprechenden Kapitels unklar.

## Perfektion aus den Staaten

Aus den USA kommt ein Textverarbeitungsprogramm mit dem Namen »Paperclip«. Entsprechend ist das ansonsten hervorragende Handbuch natürlich in Englisch abgefaßt. Auch ein deutscher Zeichensatz bleibt ein Wunschtraum. Sieht man hiervon absteht einem mit »Paperclip« allerdings ein Programm mit fast unbegrenzten Möglichkeiten zur Verfügung.

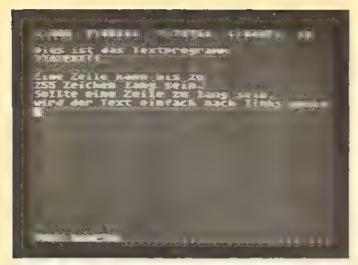


Bild 5. So kann bei dem Programm »A-Text« ein Text auf dem Bildschirm aussehen.

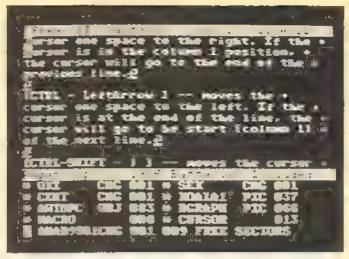


Bild 6. Das Arbeiten mit zwei Fenstern ist bei »Paperclip» ganz einfach und erleichtert so manche Problemstellung.

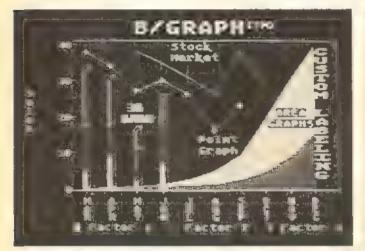


Bild 7. Eine solche graftsche Darstellung läßt sich mit »Paperclip« zu Papier bringen.



Bild 8. Ein Hauptmenü von »HomeText«. Jede Funktion läßt sich mit nur einer Taste aufrufen.

Zum Schutz vor Raubkopieren hat man sich hier übrigens etwas Besonderes einfallen lassen. Das Programm an sich ist nicht geschützt und soll sogar aus Sicherheitsgründen auf eine zweite Diskette kopiert werden. Zum Betrieb des Programms braucht man allerdings einen speziellen Schlüssel. Hierbei handelt es sich um eine kleine Box, die man am Joystickport 2 einstecken muß. Ohne diesen Schlüssel läuft gar nichts.

»Paperclip« ist zwar nur spärlich menügesteuert, jedoch fällt dies hier nicht besonders ins Gewicht. Erstens findet sich rasch jeder der einprägsamen Befehle in dem wirklich übersichtlichen Handbuch. Zweitens kann man bei Bedarf drei verschiedene Helpfiles aufrufen, in denen die Befehle kurz erläutert sind. Besonders praktisch ist hierbei, daß ein Helpfile in einem zweiten Fenster eingeblendet wird. Die wichtigsten Befehle können also parallel zum Text ständig eingesehen werden. Wer ohne Helpfile arbeitet, kann mit Hilfe der Fenster auch zwei verschiedene Texte gleichzeitig im Speicher bearbeiten (Bild 6). Das Hin- und Herschalten zwischen den Texten erledigt die SELECT-Taste.

Ist nur ein Fenster geöffnet, so wird der Text in 18 Zeilen mit je 39 Zeichen dargestellt. Die Zahl der Zeichen, die man mit dem horizontalen Scrolling nebeneinander schreiben kann, läßt sich zwischen 15 und 130 Zeichen einstellen. Wie der Text schließlich auf dem Drucker aussieht, kann man sich auch bei diesem Programm vorher ansehen. Die formatierte Darstellung erfolgt aber nicht mit 80 Zeichen Bildbreite, sondern in normaler Breite. Gegebenenfalls muß man den Text dann auf dem Bildschirm horizontal scrollen, um einen Gesamtüberblick zu erhalten.

Der Serienbrief-Ausdruck ist mit 
»Paperclip« kein Problem. Auch arbeitet dieses Programm ohne Schwierigkeiten mit bekannten Programmen wie 
»SynFile+« und »B/Graph« zusammen. Besonders interessant ist das ebenfalls auf der Diskette enthaltene Programm »HIRES«. Es handelt sich hierbei um ein Hardcopy-Programm, das Grafikbilder

von unterschiedlichstem Format ausdrucken kann. Bild 7 zeigt hierzu ein Beispiel. Die grafische Darstellung einer Statistik, die man beispielsweise mit »B/Graph« erstellt hat, kann man ausdrucken und so in einen Text ein-

Der Druckeranpassung wurde bei diesem Programm besondere Aufmerksamkeit gewidmet. Auf der Diskette befinden sich bereits über 30 fertige Files, passend zu einer Vielzahl von Druckern. Sollte sich der eigene Drucker hier trotzdem noch nicht finden, existiert ein Anpassungsprogramm, in dem man die speziellen Druckeranforderungen festhalten kann.

Zum Ausdrucken von mehreren Files kann man sich übrigens ein sogenanntes Batch-File anlegen. In solch einem File führt man einfach alle zu druckenden Textfiles auf und speichert es dann. Beim Aufruf des Batch-Files werden alle darin vorkommenden Kommandos nacheinander ausgeführt. Diese Methode wird man vor allem dann anwenden, wenn ein Text in mehreren ein-

zelnen Teilen abgelegt ist und mit einem einzigen Befehl ausgedruckt werden soll.

## 100 Prozent Menü

»HomeText« ist eigentlich keine eigenständige Textverarbeitung, sondern Teil eines Programmpakets. Unter dem Namen »HomePak« erhält man nämlich neben einer Textverarbeitung auch ein Terminal- und ein Datenverarbeitungsprogramm.

Wie schon »Paperclip« kommt auch dieses Programm aus den USA. Sowohl das Handbuch wie auch die Benutzerführung sind also in Englisch ausgelegt. Dies stört bei »HomeText« jedoch kaum. Als einzige Textverarbeitung ist »Home-Text« zu 100 Prozent menügesteuert. Allerdings ist auch hier kein deutscher Zeichensatz verfügbar, weder auf dem Bildschirm noch auf dem Drucker,

Die Stärken von »HomeText« liegen vor allem in seiner Kompaktheit und Übersichtlichkeit. Insgesamt kann man mit nur drei Menüs alle Funktionen dieses Programms ansprechen. Bild 8 zeigt eines der Menüs. Dabei versteht sich »HomeText« weniger als eigenständiges Textverarbeitungsprogramm, sondern ist vielmehr zur Zusammenarbeit mit den beiden anderen Programmen des Pakets konzipiert.

So kann man beispielsweise Daten des »Homefind«-Programms übernehmen und auf diese Weise Serienbriefe erstellen. »HomeTerm«, das DFÜ-Programm, holt sich Textdateien, die man per Akustikkoppler verdichten möchte, vom Textprogramm. Über DFÜ

empfangene Texte kann man natürlich mit »HomeText« weiterverarbeiten. Alle drei Programme befinden sich auf nur einer Diskette und können durch ein Hauptmenü schnell und bequem aufgerufen werden. Auch beim Datenaustausch gibt es durch die aufeinander abgestimmten Programme keine Probleme

Ein Nachteil von »HomeText« ist jedoch der etwas zu klein geratene Textspeicher. Er faßt nur 6620 Zeichen. Bei längeren Texten ist man also gezwungen, den Text in mehreren Files abzulegen. Die Textdarstellung erfolgt in 21 Zeilen mit 38 Spalten. Horizontales Scrollen über den rechten Bildrand hinaus ist nicht möglich. Dafür kann man sich mit dem Drucker-Preview ein Bild von dem fertigen Text machen. In skizzierter Form wird ein Blatt Papier auf dem Bildschirm dargestellt, auf dem der Text nur in Linien dargestellt ist. Dies genügt durchaus, um die äußere Form und Raumaufteilung des späteren Textes beurteilen zu können.

## Kompromisse statt optimaler Lösungen

Für welches der sechs Textverarbeitungsprogramme man sich letztlich entscheidet, hängt vor allem davon ab, inwieweit man das Programm professionell nutzen möchte. Zumindest der deutsche Zeichensatz sollte sowohl auf dem Bildschirm als auch auf einem Drucker darstellbar sein. Eine gute Menüsteuerung wird man schnell schätzen lernen, wenn man nur ab und zu Texte bearbeitet und daher umfangreiche Kontrollcodes kaum im Gedächtnis behält. Kann man sich einen Text vor dem Drucken in seiner späteren Form ansehen, erspart dies so manches Blatt Papier, Vor allem bei Briefen, in denen es auch auf eine gewisse Raumauftellung ankommt, trifft das zu. Sind häufig Tabellen zu erstellen, ist es von Vorteil, wenn die Zahl der Zeichen, die nebeneinander ohne Zeilenvorschub dargestellt werden können, möglichst groß ist. Zur professionellen Anwendung gehört natürlich auch der Serienbrief-Ausdruck, Doch wer sich die Vergleichstabelle ansieht, wird schnell feststellen, daß es das optimale Programm für den Atari nicht gibt. So bleibt dem Anwender immer nur ein Kompromiß und er wird auf das Programm zurückgreifen, das für ihn die wenigsten Nachteile aufweist.

(Wolfgang Czerny/wb)

Bezugsquellen: CompyShop, Gneisenaustr 29, 4330 M@hheim/Ruhr Tel. (0208) 497169 Münzenioher, Totzer Str 4, 8150 Hotzkirchen, Tel. (08024) 1814

## Die wichtigsten Daten der getesteten Textprogramme auf einen Blick

Atari-Schreiber Starfexter Austro-Text Paperclip

	Atari-Schreiber	ataneater	Austroliext	Paperclip	Momerext	AHEXT
Anieitung in Deutsch	jā.	ja ja	ja .	nem (Englisch)	nein (Englisch)	ja.
Datenträger	Modul oder Diskette	Diskette	Diskette	Diskette	Disketle	Modul, Diskette oder Kassette
Deutscher Zeichensatz	pà	jè	ja	nen	hein	nen
Zeichensatz wählbar	nein	ja .	nein	nein	nein	nem
Tastatur umbelegt	р	wahlwelse	nêin	nein	леи	пеіл
Printer Preview	nein	В	įa į	ja	grafisch angedeutet	пеіл
Honzontales Scrolling	nen	lä:	néin	ja	nein	j <b>a</b>
max. Zeilentänge	35	80	40	130	38	255
Druckeranpassung	teilweise voreingestelft	per Installations- programm	Installations programm	Installations-	Installations- programm	mit Steuerzeichen im Text
Centronics-interface notwendig	β	nicht unbedingt	nicht unbedingt	<u>[a</u>	jn	nicht unbedingt
Grafik-Ausdruck	nein	Grafikzeichen	nein	Hardçopy- Programm für Bilder	nein	Rigin
Menügesteuert	ja .	p	Statuszeile	teilweise	jg	Statuszerie
Formaberen auf Bildschirm	neir	įd	nur in Preview	nut in Proview	nur in Preview	nein
Qualität des Handbuches	gut	befriedigead	beinedigend	sehr gut	gut, aber knapp	befriedigend
Feinscrolling	nein	nein	ja	nen	nein	neia
Blocksatz	nut aut Drucker	auf dem Bildschirm und dem Drucker	auf dem Drucker oder in Preview	auf dem Drucker oder in Preview	nur auf dem Drucker	nur auf dem Drucker
Umfang des Textspeichers	20 448 Byte	20 000 Byte, maximal 250 Zeilen	26 762 Byte	кеine Angabe	6620 Byte	28 928 Byte
Senenbnef	ja	pen	nein	ß	ја	nein
Mathematische Funktionen	nein	p2	nem	р	nein	nen
Preis	ca, 50 Mark	64 Mark	189 Mark	149 Mark	ça. 150 Mark	49 Mark
					1	k



## 150 Befehle mit Basic XE

Wem das Atari-Basic zu unkomfortabel und zu langsam ist,
bekommt für etwa 300 Mark
Basic XE auf Modul. Es ist das
leistungsfähigste Basic für
Atari-Computer mit bis zu 128
KByte-RAM.

asic XE ist eine Werterentwicklung von Basic XL, wobei natürlich den besonderen Fähigkeiten des neuen Atari-Computers 130XE Rechnung getragen wurde. Weiterentwicklung bedeutet vor allem die Hinzunahme neuer Befehle. Dabei sah man sich allerdings scheinbar zu Kompromißlösungen gezwungen. Kam man bei Basic XL noch mit einem Modul aus, so muß bei Basic XE neben dem Modul zusätzlich noch eine Diskette als Speichermedium herhalten. Dies birgt nun gleich zwei Nachteile in sich. Zum einen kann man ohne Diskettenlaufwerk nicht mehr alle Befehle von Basic XE nutzen, zum anderen belegt das File, das die zusätzlichen Befehle enthält, immerhin 91 Sektoren auf der Diskette. Dafür sind aber auf der Masterdiskette bereits einige kleine Programme enthalten, welche die Besonderheiten von Basic XE demonstrieren sollen. Neben Modul und Diskette erhält man natürlich auch ein Handbuch, das bedauerlicherweise in englischer Sprache gehalten ist. Doch auch ohne grö-Bere Englischkenntnisse sollten die Funktionen von Basic XE aufgrund der zahlreichen Beispiellistings in diesem Handbuch verständlich werden.

Zu den Befehlen, die Bestandteil des Diskettenfiles sind, gehört auch der Befehl »FAST«. Mit dieser Anweisung gelangt man in den sogenannten »Fast Modus« von Basic XE. Ist dieses Basic bereits im normalen Modus rund doppelt so schnell wie das Standard-Atari-Basic, so erhöht sich im »Fast Modus« die Verarbeitungsgeschwindigkeit von Programmen nochmals um den Faktor zwei. Die insgesamt also etwa vierfache Steigerung der Geschwindigkeit reicht so oftmals schon aus, um Programme zu schreiben, die sonst ohne Maschinensprache-Unterstützung zu langsam

wären. Ebenfalls auf Diskette befindet sich ein File, das nur vom Atari 130XE oder aufgerüsteten Atari 800XL-Computern verarbeitet werden kann. Mit dem Befehl »EXTEND« spricht man die zusätzlichen 64 KByte an, die dem 130XE zur Verfügung stehen. Dieser Befehl verlegt das momentan im Speicher befindliche Programm in diese 64 KByte und schafft so Platz für andere Programme und Daten, Der Zusatzspeicher ist in vier Blocks von jeweils 16 KByte aufgeteilt. Jedem Block ist eine Nummer von 0 bis 3 zugeordnet. Ein spezieller POKE-Befehl erlaubt dann unter Angabe der Blocknummer, einzelne Daten in den Zusatzspeicher zu verlegen.

## Player/Missiles mit Komfort

Zu den neuen Befehlen, die gegenüber Basic XL hinzugekommen sind, gehören auch eine Reihe von Anweisungen, die die Player/Missile-Programmierung betreffen. War man im normalen Basic noch gezwungen, eine Vielzahl verschiedener Adressen zu kennen, um Player oder Missiles anzusprechen, so erübrigt sich dies in Basic XE fast vollständig. Für nahezu jede Manipulation existiert ein eigener Befehl, »PMMOVE« beispielsweise veranlaßt eine beliebige Richtungsänderung eines Plavers oder Missiles. Auch die Kollisionsabfrage läßt sich mit einem einzigen Befehl mit dem Namen »BUMP« ausführen. Als recht nützlich erweist sich die Anweisung »PMCLEAR«, mit der sich der Speicherbereich, in dem ein Player oder eine Missile untergebracht ist, schnell und einfach löschen

Die Stringverarbeitung, die im normalen Basic etwas stiefmütterlich behandelt wird, ist nun wesentlich vereinfacht und damit auch komfortabler geworden. Es fängt damit an, daß man mit Basic XE auch indizierte Springfelder dimensionieren kann. Mit der Anweisung »DIM A\$ (3,10)« beispielsweise werden drei Felder mit der Länge 10 erzeugt, die sich dann mit »A\$(1)«, »A\$(2)« und »A\$(3)« aufrufen lassen, Einzelne Teile dieser Felder bearbeitet man dann mit den Befehlen »LEFT\$«, »RIGHT\$«, »MID\$« oder »FIND\$«. So gibt »FIND\$« beispielsweise an, ob und wo sich einzelne Zeichen in einem String befinden. Das Sortieren von Arrays ist mit Basic XE ebenfalls kein Problem mehr, Hierzu verwendet man die Befehle »SORTUP« und »SORTDOWN«. Numerische oder alphanumerische Felder lassen sich also in aufsteigender oder abfallender Folge sortieren. Dabei kann man entweder das gesamte Feld oder nur ausgewählte Teilbereiche erfassen.

## Programmieren, fast wie in Pascal

Auffallend sind vor allem auch Pascalähnliche Elemente, die in Basic XE auftauchen. Unterprogramme lassen sich zum Beispiel als Prozeduren gestalten. Diese können mit »CALL« aufgerufen und mit »EXIT« wieder verlassen werden, Innerhalb von Prozeduren kann man dann mit lokalen Variablen arbeiten. Das heißt, solche Variablen können zwar namensgleich mit anderen im Programm verwendeten Variablen sein, sind mit diesen jedoch nicht identisch. Ebenfalls von Pascal entlehnt ist sowohl die »WHILE-ENDWHILE«-Schleife als auch die bedingte Verzweigung JF-ELSE-ENDIF«. Auf diese Weise lassen sich vor aliem die uneleganten und zur Unübersichtlichkeit beitragenden »GOTO«-Anweisungen weitgehend aus einem Programm verbannen. Mit Basic XE ist es also möglich, bis zu einem gewissen Grad strukturiert zu programmieren und so eine wesentlich bessere Übersichtlichkeit eines Programms zu erzielen. Dies wird zudem durch die Tatsache unterstützt, daß ein Listing in Basic XE formatiert ausgegeben wird.

```
118 Rea BEISPIEL-LISTING
120 Rom mit Basic XE
130 Res -
140 Graphics 0
150 ? "RATEN SIE EINE ZAHL"
160 ? "ZWISCHEN 1 UND 180"
170
190 While Versuch<>Zahl
200 ? "WIE LAUTET IHR VERSUCH ?";
218
         Input Versuch
         If Versuch<Zahl
? "DIE ZAHL IST ZU KLEIN"
220
240
            If Versuch>Zahl
238
                  "DIE ZAHL IST ZU GROSS"
            Endif
270
280 Endif
290 Endwhile
300 2 : 2 "GE
310 End
              "GEFLINDEN"
```

Listing. Ein typisches Basic XE-Programm.

Schleifen und »IF«-Anweisungen werden also automatisch eingerückt und so optisch hervorgehoben. Ein typisches Basic XE-Programm zeigt das Listing 1.

Viele der neuen Routinen sind dem geübten Programmierer bereits von oft benötigten Maschinensprache-Unterprogrammen her bekannt. So verschiebt man Speicherblöcke mit dem Befehl »MOVE«. Dadurch kann man beispielsweise den Inhalt des Bildschirmspeichers in einem String ablegen oder den Atari-Zeichensatz kopieren. Die automatische Zeilennumerierung bei der Programmeingabe und die Neunumerierung eines bestehenden Programms sind ebenfalls ohne Maschi-

nenhilfsroutinen durchführbar. Hierzu dienen die Befehle »NUM« und »RENUM«. Benötigt man eine Liste aller im Programm verwendeten Variablen, so erlaubt das der Befehl »LVAR«. Zur schnellen Umwandlung von dezimalen in hexadezimale Zahlen benutzt man den »HEX\$«-Befehl. Nützlich ist der Befehl »SET«. Mit ihm lassen sich eine Reihe von grundsätzlichen Parametern allgemeiner Art ändern. So ermöglicht dieser Befehl beispielsweise die Auswahl, mit welchem Zeichen eine Input-Anweisung den Benutzer zur Eingabe auffordert. Die Funktion der »BREAK«-Taste kann auch auf diese Weise unterbunden werden. Selbst das Dimensionieren von Strings erübrigt sich mitunter. Bis zu einer Größe von 255 Zeichen, abhängig von der jeweiligen »SET«-Einstellung, werden Strings nämlich automatisch dimensioniert. Insgesamt können 16 verschiedene Parameter mit diesem Befehl beeinflußt werden. Die Einstellung der Parameter fragt man mit dem Befehl »SYS« ab.

Hilfreich bei der Fehlersuche ist der \*TRACE«-Befehl, Vor der Ausführung einer Programmzeile erscheint die jeweilige Zeilennummer auf dem Bildschirm. So kann man leicht kontrollieren, ob sich ein Programm in den gewünschten Bahnen bewegt. Die Textausgabe erfolgt wahlweise invers oder normal, was einfach durch die beiden Befehle »INVERSE« und »NOR-MAL« gesteuert wird. Man kann einen auszugebenden Text aber auch noch formatiert darstellen. Dies geschieht mit dem »PRINT USING«-Befehl.

Auch einige DOS-Funktionen finden sich in Basic XE wieder. So läßt sich das Inhaltsverzeichnis einer Diskette zum Beispiel mit dem Befehl »DIR« abrufen. Für das Löschen und Umbenennen von Files auf Diskette existieren eigene Befehle. Neben den gewohnten Befehien »GET«, »PUT«, »PRINT« und »INPUT«, die zum Lesen und Schreiben von Daten auf Diskette verwendet werden, stellt Basic XE zu diesem Zweck einige neue Befehle zur Verfügung. Sie zeichnen sich vor allem durch eine höhere Ausführungsgeschwindigkeit aus. Mit »BPUT« und »BGET« schreibt beziehungsweise liest man ganze Speicherblöcke von Diskette. Auf diese Weise lassen sich beispielsweise Grafikbilder schnell und einfach von Diskette laden oder speichern. »RPUT« und »RGET« sind für das Speichern und Lesen von Records mit fester Länge verantwortlich. Dabei kann ein Record sowohl aus einem String wie auch einer Gleitkommazahl bestehen. Schließlich gibt es noch die Befehle »BLOAD« und »BSAVE«, die man für die Behandlung von Binärfiles im »DOS LOAD«-Format benutzt. Normale Maschinensprachefiles können also von Basic aus geladen und gestartet werden.

## Gut, aber teuer

Die Qualität der Befehle und der Befehlsumfang (Basic XE umfaßt zirka 150 Befehle) lassen kaum Wünsche offen. Lediglich das Laden zusätzlicher Befehle von Diskette scheint keine glückliche Lösung zu sein. Das entscheidende Kaufhindernis dürfte allerdings der Preis von Basic XE sein. Mit etwa 300 Mark ist es wohl eines der teuersten Programme, die derzeit für Atari-Computer angeboten werden. Der Preis fällt um so mehr ins Gewicht, wenn man bedenkt, daß beispielsweise Turbo-Basic XL bei ähnlichen Leistungsmerkmalen doch wesentlich billiger zu haben ist.

(Wolfgang Czerny/wb)

Sezugsquelle CompyShop, Gnessenaustr 29, 4330 Mühlheim/Ruhr, Tel. (0208) 497169

## DOS-Parade

Wichtigstes Hilfsmittel zum Arbeiten mit einer Diskettenstation ist das DOS. Doch welche der verschiedenen DOS-Versionen soll man wählen?

OS ist die Abkürzung für Disk Operating System oder auf deutsch Disketten Betriebssystem. Betreibt man ein oder mehrere Diskettenlaufwerke am Atari, so kommt man ohne ein solches DOS nicht aus. Die zum Betrieb eines Laufwerks nötigen Routinen sind nämlich nicht im eigentlichen Betriebssystem des Atari verankert, sondem müssen stets von Diskette geladen werden. Dies hat den Vorteil, daß man sich, abhängig vom verwendeten Laufwerk und den speziellen Bedürfnissen, das jeweils optimale DOS heraussuchen kann.

Von Atari selbst gibt es nun schon die vierte offizielle DOS-Version. Angefangen hat es mit »DOS 1.0«. Diese erste

Version, die zusammen mit dem Atari-Laufwerk 810 Ende 1979 erschien, war noch mit etlichen Nachteilen und kleineren Fehlern behaftet. So gab es beispielsweise noch kein »AUTORUN. SYS«-File, um Programme nach dem Einschalten des Computers automatisch zu laden und zu starten. Andererseits konnte zum Kopieren von Programmen und Disketten nur ein relativ kleiner Puffer und nicht der gesamte RAM-Speicher verwendet Auch war der wahlfreie Zugriff auf einzelne Byte in bestimmten Sektoren mit den NOTE- und POINT-Befehlen noch nicht implementiert. Es verwundert also kaum, daß Atari bereits Anfang 1980 eine neue Version, nämlich »DOS 2.0« (Bild 1) herausbrachte.

Dieses DOS war nun frei von den Fehlern und den gröbsten Nachteilen der ersten Version. Allerdings ging man auch hier Kompromisse ein. War DOS 1.0 noch ständig komplett im Speicher vertreten, teilte man DOS 2.0 in zwei getrennte Files auf. Zum einen in das DOS.SYS-File, das durch den Boot-Vorgang in den Speicher des Atari gelangte, zum anderen in das DUP.SYS-File, das nur bei Bedarf durch den DOS-Aufruf von Basic aus geladen wurde. Vorteil dieser Methode ist die Einsparung von wertvollem Speicherplatz. Nur

noch rund 2 Kilobyte des RAM-Speichers gingen durch die Belegung des DOS.SYS-Files verloren. Die wichtigsten Funktionen, wie das Speichern und Laden von Programmen waren dabei noch durchführbar. Auch konnte man mit speziellen XIO-Routinen beispielsweise Disketten formatieren oder Files auf Diskette löschen. Spätestens aber zum Kopieren von Programmen oder zur Ausgabe des Disketteninhaltes auf dem Bildschirm, mußte man das DUP.SYS-File aufrufen. Dabei wurden im Arbeitsspeicher befindliche Basic-Programme jedoch überschrieben. Zur Rettung des Speichermhaltes mußte man sein Programm daher entweder vor jedem DOS-Aufruf speichern oder aber ein sogenanntes MEM.SAV-File anlegen, in welches das Programm vor dem Laden des DUP.SYS-Files automatisch gespeichert wurde. Dies ist aber zeitaufwendig und bedeutet zusätzlichen Platzverbrauch auf der Diskette. Abgesehen davon konnte man mit diesem DOS jedoch zufrieden sein, und es ist wohl heute noch das meistverbreitete DOS für Atari-Computer.

Als 1983 ein verbessertes Atan-Laufwerk 1050 erschien, entschloß man sich bei Atari auch für eine neue DOS-Version, dem DOS 3.0 (Bild 2 zeigt das Hauptmenü von DOS 3.0).



Bild 1. Das Hauptmenü zu DOS 2.0. Sämtliche Disketten-Operationen lassen sich von hier ausführen. Da Disketten allerdings mit maximal 720 Sektoren zu formatieren sind, ergibt sich eine Gesamtspeicherkapazität von rund 88 KByte.

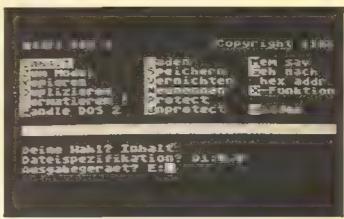


Bild 2. DOS 3.0 präsentiert sich mit einem etwas anders aufgebauten Menü. Als Besonderheit gibt es hier die »Hilfen«-Funktion, mit der man sich erklärende Texte zu den einzelnen Menufunktionen direkt auf den Bildschirm holen kann.

Der Hauptgrund hierfür war eine spezielle Eigenschaft des neuen Laufwerks. Neben der normalen Schreibdichte, die auch das Laufwerk 810 beherrscht, kann das 1050er Laufwerk in einer höheren Schreibdichte (enhanced density) arbeiten. Dies ist nicht mit der doppelten Schreibdichte (double density) zu verwechseln. Denn dies würde beim Atari 1050-Laufwerk bedeuten, daß pro Sektor statt 128 Byte 256 Byte Platz fänden. Bei gleicher Sektorenzahl würde man also die doppelte Speicherkapazität erreichen.

#### **DOS 3.0**

Erhöhte Schreibdichte bedeutet nur, daß statt 18 jetzt 26 Sektoren pro Spur angelegt werden können. Die Anzahl erhöht sich also von 720 auf 1040. Praktisch verfügt man nur über 1023 Sektoren, da das Laufwerk die übrigen 17 nicht mehr verwalten kann.

Bei DOS 3.0 kam man auf die kuriose Idee, jeweils acht Sektoren zu einem Block von 1024 Byte zusammenzufassen. Dies erleichtert dem Directory zwar die Verwaltungsarbeit, hat aber zwei entscheidende Nachteile. Einmal wird dadurch in der Regel enorm viel Platz verschwendet Die Mindestlänge eines Programms auf der Diskette beträgt nun nämlich einen Block beziehungsweise acht Sektoren. Selbst ein Programm mit einer Länge von nur 10 Byte belegt damit einen ganzen Block, also 1024 Byte. Außerdem ist dieses Format unglücklicherweise nicht kompatibel zu DOS 2.0. Eine mit DOS 3.0 formatierte Diskette kann also nicht unter DOS 2.0 gelesen werden. Umgekehrt sind die alten Atari-Laufwerke 810 nicht mehr in der Lage, DOS-3.0 Disketten zu verarbeiten. Ein unter DOS 2.0 erstelltes File kann zwar auf DOS 3.0-Format umgewandelt werden, umgekehrt ist dies jedoch nicht durchführbar, Zum Trost sei aber gesagt, daß zu diesem Zweck bereits Utilities in verschiedenen Zeitschriften erschienen sind.

Bild 3. Wer DOS 2.0 bereits kennt, kommt mit DOS 2.5 schnell zurecht. Der einzige Unterschied: Mit »I« formatiert man eine Diskette im enhanced Modus und mit »P« erhält man ein Diskettenformat identisch mit DOS 2.0.

Trotz dieser Nachteile mußte man bis zur Einführung des neuen Atari 130XE warten, bis die vorläufig letzte Version. nämlich DOS 2.5, erschien, DOS 2.5 kann man getrost als das beste DOS bezeichnen, das von Atari bisher geliefert wurde (Bild 3). Es unterstützt die erhöhte Schreibdichte des 1050er Laufwerks. Die Diskettenverwaltung durch Blocks ist aufgehoben worden, so daß die Filekompatibilität zur Version 2.0 wieder hergestellt ist. Es sind also einzelne Sektoren angesprochen, deren Anzahl aber gestiegen ist. Dies hat auch den Vorteil, daß man Disketten mit erhöhter Schreibdichte wenigstens zum Teil mit dem alten Laufwerk 810 lesen kann. Lediglich die Sektoren zwischen 720 und 1023 bleiben dem 1050er Laufwerk vorbehalten. Sind mehr als 999 Sektoren frei, so werden diese übrigens nicht mehr angezeigt. Dadurch sollte man sich jedoch nicht irntieren lassen. Natürlich unterstützt DOS 2.5 auch die 128 KBvte des 130XE. Vor allem kann man im zusätzlichen - 64 KByte umfassenden - RAM-Speicher eine sogenannte RAM-Disk anlegen, die wie ein Diskettenlaufwerk mit der Laufwerknummer 8 angesprochen wird. Man verfügt dann über rund 500 »Sektoren«, deren Inhalt quasi ohne Zeitverlust geladen oder beschrieben werden kann. Außerdem hat man noch einige andere neue Fähigkeiten in das DOS aufgenommen, die zwar nicht im Menü aufgeführt sind, jedoch über ».COM«-Files ausgeführt werden können. Versehentlich gelöschte Programme wiederherzustellen, sofern sie nicht bereits überschrieben wurden, ist nun kein Problem mehr, DOS 3.0-Files lassen sich in das DOS 2.5-Format umwandeln. Auch kann man AUTO-RUN.SYS-Files zum automatischen Programmstart von Basic-Programmen



Bild 4. DOS XL befindet sich ständig im RAM-Speicher. Gibt man von Basic aus »DOS» ein, gelangt man ohne Diskettenzugriffe zur Kommandozeile. Befindet sich zu dem Zeitpunkt ein Basic-Programm im Speicher, wird es nicht gelöscht.

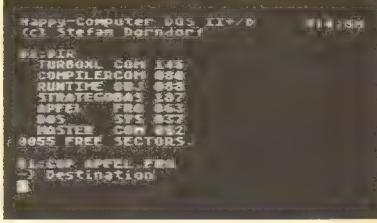


Bild 5. Happy-DOS (Listing des Monats in Ausgabe 3/86) ist ähnlich aufgebaut wie DOS XL. Es liegt ebenfalls stets im Speicher vor. Als besonderen Leckerbissen bietet Happy-DOS noch eine RAM-Disk, die insgesamt 12 KByte umfaßt.

erzeugen. Da für jedes vorgesehene Laufwerk 128 Byte Speicherplatz im DOS reserviert werden, ist es ein leichtes, die Anzahl der verwendeten Laufwerke an die eigene Konfiguration anzupassen. So lassen sich auch noch einige Byte einsparen. Etwas störend wirkt sich die Anordnung des Directory-Eintrags auf einer DOS 2.5 formatierten Diskette aus. Da dieser Eintrag nicht in einem Stück, sondern verteilt auf der Diskette plaziert ist, macht sich der Schreib-/Lesekopf des Laufwerks immer wieder durch lautstarke Positionswechsel bemerkbar.

Natürlich gibt es auch DOS-Versionen von anderen Herstellern als Atari, beispielsweise DOS XL 2.3 von OSS. Unter dieser DOS-Version laufen so bekannte Programme wie »Action«, »MAC/65« oder »Basic XL«. Obwohl dies der Name nicht vermuten laßt, ist DOS XL für alle Atari-Computer verwendbar, da drei verschiedene Versionen auf der Master-Diskette vorhanden sind - passend zu der jeweiligen Speicherkonfiguration. Die Kommandozeile mit einem teilweise aufgelisteten Directory zeigt Bild 4. Neben der normalen Schreibdichte wird bei DOS XL sogar die doppelte Schreibdichte unterstützt. Um dies zu nutzen, muß man allerdings entweder das Atari 1050-Laufwerk hardwaremäßig erweitem (wie im Artikel »Floppy-Speeder« in diesem Sonderheft beschrieben), oder man verwendet ein Laufwerk eines Fremdherstellers. Die Speicherkapazität einer Diskette steigt dann allerdings von 90 KByte unter normaler Schreibdichte oder 130 KByte bei erhöhter Schreibdichte auf stolze 176 KByte. Dieses Format ist dann natürlich von den Atari-DOS-Versionen nicht mehr lesbar. Verzichtet man aber auf doppelte Schreibdichte, so besteht Filekompatibilität

zwischen DOS XL und DOS 2.0 beziehungsweise DOS 2.5.

DOS XL verzichtet auf eine Trennung zwischen DOS.SYS und DUP.SYS-File. Das heißt aber nicht, daß keine weiteren Files nachgeladen werden. Man unterscheidet hier zwischen Kommandos. die das DOS ohne weiteres Zutun ausführt und Anweisungen, die nur durch Nachladen eines entsprechenden Files ausgeführt werden. Dies hat den Vorteil, daß das eigentliche DOS nur wenig Speicherplatz im RAM belegt, zumal es recht geschickt hinter dem Betriebssystem ROM abgelegt ist. Im eigentlichen Basic-RAM-Bereich befindet sich nur noch eine Sprungtabelle, die ungefähr 500 Byte beansprucht. Der Nachteil dieser Methode besteht jedoch darin, daß die extern gelagerten Files relativ viel Platz auf der Diskette verbrauchen. Es ist also ratsam, auf die jeweilige Arbeitsdiskette nur diejenigen DOS-Files zu kopieren, die man unbedingt benötigt. Auch das Menü von DOS XL gehört zu diesen Files, denn im Normalfall arbeitet DOS XL nur mit einer Kommandozeile. Hat man alle DOS-Befehle im Kopf, läßt es sich auf diese Art sogar wesentlich schneller arbeiten als mit einem Menü.

Eine Besonderheit stellt die sogenannte Batch-Verarbeitung dar. Man
kann hierbei beliebigen Text und DOSAnweisungen in einem File zusammenstellen. Versehen mit dem Extender
».EXC« besitzt man dann ein File, das
man von der Kommandozeile aus aufrufen kann. Das File »STARTUP.EXC«, das
dem herkömmlichen »AUTORUN.SYS«
entspricht, ist ein Beispiel für so ein
Batch-File. Auch häufig verwendete
Befehlssequenzen lassen sich auf diese Art zeitsparend ausführen. Neben
den kommerziellen Programmen kann
man natürlich auch auf die verschieden-

sten DOS-Versionen zurückgreifen, die beispielsweise als Public Domain-Programme erhältlich sind, oder als Listings in Zeitschriften veroffentlicht werden. Ein Beispiel ist das in Happy-Computer, Ausgabe 3/86 als Listing des Monats abgedruckte Happy DOS (Bild 5). Ähnlich wie bei DOS XL wird nur mit einer Kommandozeile gearbeitet. Jedoch sind alle Befehle im DOS.SYS-File enthalten, so daß ein Nachladen von Files entfällt. Trotzdem beansprucht das DOS-File nur 37 Sektoren auf Diskette. Zusätzlich ist noch eine 12 KByte große RAM-Disk integriert, die auch mit den normalen Atari-Computern angesprochen werden kann. Es läßt auch zu, wie in DOS 2.5. Disketten mit erhöhter Schreibdichte zu formatieren.

Die Entscheidung für ein bestimmtes DOS solite vor allem von der allgemeinen Filekompatibilität abhängig gemacht werden. Speziell DOS 3.0 sollte man trotz seiner relativ guten Benutzerführung also vermeiden. Kommt es auf äußerste Platzersparnis im RAM an, oder besitzt man ein Laufwerk für doppelte Schreibdichte, so ist man mit DOS XL am besten bedient. Allgemein kann man jedoch sagen, daß DOS 2,5 von Atari das derzeit interessanteste Disketten-Betriebssystem darstellt. Arbeitet man mit Basic-Versionen wie zum Beispiel Turbo-Basic XL, kann man auf den Sprung ins DOS-Menü und somit auf das Laden der DUP.SYS-Datei auch meist verzichten. Einige DOS-Funktionen sind dann námlich bereits als Basic-Befehle vorhanden. Die integrierten Utilities und die Diskettenformatierung in normaler oder erhöhter Schreibdichte, lassen also gute Zukunftsaussichten für DOS 2.5 vermuten.

(Wolfgang Czerny/ub)

Bezugsquelle: CompyShop, Gnersenaustr 29, 4330 Müntheim/Ruhr Tel. (0208) 497169



## MAC/65: ein Assembler sprintet los!

Schnell stößt man in Basic an die Grenzen der erreichbaren Geschwindigkeit. Selbst eine optimale Programmierung verleiht Basic keine Flügel. Wer es eilig hat, sollte auf einen Assembler zurückgreifen.

s gibt sicherlich verschiedene Gründe dafür, auf Assembler-Programmierung umzusteigen. Vorrangig dürfte aber die extreme Geschwindigkeitssteigerung über einer Hochsprache, wie beispielsweise Basic, sein. Auch diejenigen, die sich lieber mit der Hardware eines Computers beschäftigen, greifen vorzugsweise auf Assemblerprogrammierung zurück. Denn nicht selten ist es nur dann möglich, bestimmte Bauteile im Computer anzusprechen und zu programmieren. Wer beispielsweise den Antic, also den speziellen Grafikprozessor im 800XL/130XE, optimal nutzen möchte, kommt an Assembler nicht vor-

Übrigens sind Assembler und Maschinensprache nicht dasselbe. Wer nämlich in Maschinensprache programmiert, der ordnet jeder einzelnen Speicherzelle im RAM-Speicher einen bestimmten Hexadezimalcode zu. Es wird also nur mit Zahlenwerten gearbeitet. In Assembler andererseits programmiert man mit leichter verständlichen Befehlen, den sogenannten Mnemonics. Mit dem Editor eines Assemblers gibt man Befehle wie LDA oder CMP ein. Diese werden dann vom Assembler in Maschinencode umgewandelt. Erst wenn dieser Code im Speicher vorliegt, kann der Microprozessor die Befehle abarbeiten.

Aber nicht allein Zeitgewinn oder auch Platzerspamis sind gute Gründe für die Anwendung von Maschinensprache. Um beispielsweise die Fähigkeiten des Atari-Betriebssystems auszureizen, muß man zumindest einige Programmteile in Maschinensprache schreiben.

Ob man eine kleine Unterroutine für ein Basic-Programm oder ein ganzes Programm vollständig in Maschinensprache schreiben will, ein Assembler ist auf jeden Fall eine Johnende Investition

Von der Firma OSS (Optimized Systems Software) gibt es für etwa 250°

Mark das Programmodul »MAC/65«. Es enthalt alle wesentlichen Bestandteile eines Assemblers – nämlich einen zeitennummernorientrerten Editor, ähnlich wie in Basic, den eigentlichen Assembler und einen Monitor, über den man seine Maschinensprachprogramme austesten kann.

Die Modulform bietet mehrere wichtige Vorteile gegenüber einem Programm auf Diskette oder Kassette. So kann man beispielsweise nach Lust und Laune mit jeder Kombination von Diskettenlaufwerken arbeiten, da man ja in der Wahl des Diskettenbetriebssystems völlig frei ist. Außerdem muß man nicht nach einem eventuellen Systemabsturz, der ja bei Programmierung auf Maschinenspracheebene relativ schnell passieren kann, den Assembler neu laden. Darüber hinaus hat es OSS durch Umschaltung zwischen ver-Speicherbänken schiedenen schafft, den Speicherplatzbedarf von MAC/65 auf 8 KByte zu beschränken.

Nun aber zum eigentlichen Programm - beginnen wir mit dem Editor:

Genau wie im normalen Atari-Basic hat man es mit einem zeilennummernorientierten Editor zu tun. Er ist jedoch um einige sehr wichtige Kommandos erweitert. So kann man mit FIND bestimmte Programmteile suchen, mit REP ersetzen, mit DEL Programmbereiche löschen und mit RENUM das gesamte Programm neu durchnumerieren Auch ein Befehl zur automatischen Erzeugung von Zeilennummern fehlt nicht. Eine praktische Sache ist, daß MAC/65, genau wie Atan-Basic, die Programmzeilen unmittelbar nach Eingabe auf ihre Richtigkeit überprüft und sofort einen eventuellen Fehler anzeigt.

Nun zum Assembler, den man mit dem Kommando ASM startet. Dabei kann man noch angeben, woher – falls nicht aus dem RAM-Speicher – der Programmtext genommen werden soll, wohin die Ausgabe gehen soll und wohin der erzeugte Maschinencode geschrieben wird.

Selbstverständlich versteht MAC/65 alle normalen 6502-Mnemonics.

Datenfelder kann man einfach mit dem Befehl ». Byte« einfügen – wobei man entweder dezimale, hexadezimale oder ASCII-Darstellung wählen (und mischen) kann. Sehr nützlich ist dabei die Möglichkeit, zu allen errechneten Werten eine Konstante zu addieren.

Weitergehende Befehle in dieser Richtung sind ».SBYTE« (Zeichen in interner Darstellung), ».WORD« (Doppelbyte), ».DBYTE« (Doppelbyte in umgekehrter Schreibweise), ».CBYTE« (Text mit invertiertem letzten Zeichen) und ».FLOAT« (Fließkommazahl). Der »MAC/65«-Assembler erlaubt auch das Arbeiten mit Markierungen (Labels) innerhalb eines Programms. Die Länge einer Zeichenkette läßt sich zum Beispiel folgendermaßen berechnen:

1000 TEXT .BYTE "Happy Computer"
1010 TEXT\_LEN \* \*-TEXT

Auch zu Multiplikationen, Divisionen oder Bitoperationen in Ausdrücken ist der »MAC/65« fähig.

Nützlich ist die konditionelle Assemblierung. Mit Hilfe dieser Funktion lassen sich bei der Assemblierung bestimmte Programmteile von irgendeiner Bedingung abhängig machen. Hat man beispielsweise folgende Variablen festgelegt.

1000 DEUTSCH=1

1010 VERSION-DEUTSCH

dann kann man später mrt

2000 . IF VERSION = DEUTSCH

2010 .BYTE Bitte Diskette

einlegen"

2020 .ELSE

2030 .BYTE "Please insert disk"

2040 .ENDIF

zwischen den verschiedenen Fällen unterscheiden. So braucht man keine verschiedenen Quelitexte, um unterschiedliche Versionen eines Programms zu schreiben.

In diesem Zusammenhang ist die Funktion ».REF« wichtig. Mit ihr kann man innerhalb einer ».IF«-Anweisung feststellen, ob ein bestimmtes Label innerhalb des Textes bereits benutzt wurde. Dies gestattet Unterroutinen nur dann mit einzubinden, wenn sie tatsächlich gebraucht werden. Zusammen mit der ».INCLUDE«-Funktion, mit der sich Programmteile von Diskette einbinden lassen, kann man auch Unterprogramme aus Bibliotheken nutzen.

Ein wichtiges Merkmal von »MAC/65« ist die Fähigkeit, Makros zu verarbeiten (daher auch der Name MAC/65). Mit Hilfe von Makros kann man mehrere normale Assemblerbefehle unter einem Wort zusammenfassen. Es lassen sich auch Parameter übergeben und verarbeiten. Ein Beispiel für eine einfache Makrodefinition: 1000 .MACRO JEQ 1010 BNE ?JEQ 1020 JMP %1 1030 ?JEQ 1040 . ENDM

Zunächst wird hierbei der Name des Makros festgelegt. »JEQ« soll hier einen Befehl simulieren, der im normalen 6502-Befehlssatz nicht vorhanden nämlich einen konditionellen Sprungbetehl zu einer absoluten Adresse (in diesem Beispiel, falls die Abfrage zutrifft). Falls nicht (Branch Not Equal), wird an das Ende der Makrodefinition verzweigt. Das Programm läuft also beim nächstfolgenden Befehl weiter. Im anderen Fall wird ein Sprung an die als erster Parameter angegebene Adresse durchgeführt. Der Aufruf könnte dann so aussehen:

2000 LDA ZAEHLER 2010 CMP # HOECHSTWERT 2020 JEQ KEINE\_VERAENDERUNG

Hier kann man übrigens auch sehr gut sehen, daß Assemblerprogramme nicht unbedingt schwer lesbar sein müssen. Immerhin kann man, soweit es der Speicherplatz erlaubt, die Labelnamen behebig lang und damit ziemlich selbsterklärend gestalten.

Ein weiteres Beispiel wäre ein Makro zur Addition von 16-Bit-Werten, für die es bekanntlich beim 6502 keinen eigenen Befehl gibt:

1000 .MACRO AIW ; Add Immediate to Word

1010 .IF %2=2 :Eins addieren? 1020 INC %1

1030 BNE ?NO INC

1040 INC \$1+1 1050 NO\_INC

1060 .ELSE

1070 LDA %1

1080 CLC

1090 ADC # < %2

1100 STA %1

1110 LDA \$1+1

1120 ADC #> %2

1130 STA \$1+1

1140 .ENDIF

1150 .ENDM

Hier wird zuerst geprüft, ob das zweite Argument, also die Zahl, die addiert werden soll, den Wert 1 enthält. In diesem Fall kann der etwas kürzere Weg über die Inkrementierung der beiden Bytes gegangen werden. In allen anderen Fällen (».ELSE«-Abfrage) werden der Reihe nach zunächst das niederwertigere und dann das höherwertigere Byte addiert. Man beachte, daß Makros auch eine Optimierung des Programmtextes vornehmen können, wenn sie sinnvoll konditionell definiert sind. Obwohl im Programmcode nun jedes Mittel zur Kürzung des Codes vorgenommen werden kann, sieht der tatsächliche Quelltext wesentlich übersichtlicher aus - und ist zudem noch

kürzer! Was will man mehr? Auch hier ein Beispiel für die Benutzung dieses Makros:

2000 AIW PNT, 7680 Man vergleiche dies mit: 2000 LDA PNT

2020 ADC # < 7680

2010 CLC

2030 STA PNT

2040 LDA PNT+1

2050 ADC # > 7680

2060 STA PNT+1

Mit Makros kann man aber nicht nur oft benutzte Kombinationen von Befehlen, sondern auch Befehlsfolgen, die in einem eigenen Programm regelmäßig auftreten, verkürzen. Auch zum Aufbau von Datentabellen sind Makros nützlich, da sich auch die übergebenen Parameter direkt in Daten umwandeln lassen

Ein weiterer Pluspunkt von Makros ist, daß es Umsteigem von Basic erleichtert wird, in Assembler zu programmieren - Makrobibliotheken, die Makros wie »PRINT«, »INPUT« oder »GRAPHICS« enthalten, gibt es zur Genüge.

Kommen wir zum eigentlichen Assemblerteil von »MAC/65«. Hier lassen sich für den Quelltext, das Listing und den erzeugten Code beliebige Peripheriegeräte festsetzen. Dabei kann man den Quelltext auch direkt aus dem Speicher übernehmen. Oder man legt den generierten Maschinencode direkt im RAM-Speicher ab. Dabei muß man natürlich darauf achten, den im Speicher befindlichen Programmtext nicht zu überschreiben. Das Wichtigste ist natürlich die Assembliergeschwindigkeit. Dazu ein Beispiel: Für etwa 100 KByte Quelitext (wobei zu beachten ist, daß MAC/65 die Befehle in Tokens verwandelt und daher relativ wenig Platz für Quelltexte verbraucht) benötigt der Assembler etwa drei Minuten. In diesem Zeitraum wurde der gesamte Quelitext von Diskette gelesen und der erzeugte Code wiederum auf Diskette geschrieben. Hier darf man nicht verschweigen, daß die meiste Zeit beim Assemblieren durch Zugriffe auf externe Speichermedien verlorengeht. Deshalb lassen sich bessere Ergebnisse erzielen, wenn man auf dem Atari 130 XE mit der RAM-Disk unter DOS 2.5 arbeitet.

Ein anderer wichtiger Gesichtspunkt ist, daß der »MAC/65«-Assembler, nachdem er mehrmals verbessert wurde, als fast fehlerfrei bezeichnet werden kann. Dies bestätigt mehrjährige Erfahrung. Was wäre schlimmer als ein Assembler, der bereits beim Assemblieren Fehler in den Maschinencode bringt?

Doch der Assemblerteil Von »MAC/65« hat noch einen weiteren Leckerbissen zu bieten: Für die Ausgabe des Protokolls beim Assemblieren auf Druckern stehen vielfältige Funktionen zur Verfügung. Man kann sogar den Inhalt und die Form des Listings beeinflussen. So ist es zum Beispiel kein Problem, die Erzeugung des Listings für bestimmte Teile des Programms abzuschalten oder bei Makros alle nicht wichtigen Teile wegzulassen, Außerdem lassen sich die Seitenlänge und -breite sowie der Steuercode für Seitenvorschub festlegen. Diese Angaben befähigen den »MAC/65«, ein übersichtliches Listing mit Seitennumerierung und Seitenüberschriften zu erzeugen. Wahlweise kann auch noch eine Cross-Reference-Liste ausgestellt werden. So könnte in alphabetischer Reihenfolge für jedes benutzte Label nicht nur der erzeugte Wert, sondem auch jedes Auftauchen im Programm mit Seiten- und Zeilennummer angegeben werden. Eine sehr nützliche Funktion, um zeitaufwendige Fehlersuche zu ersparen.

Bestandteil Der dritte »MAC/65«-Assembler ist der Monitor. Er nennt sich »DDT« (Dunions Debugging Tool), da er durch Eingabe von DDT aufzurufen ist. Hier sieht man sofort den ersten großen Unterschied zu konventionellen Debuggern. Statt einen normalen Textbildschirm zu verwenden, erzeugt er ein spezielles Bildschirmformat. Dieses ist sehr übersichtlich und informativ und beinträchtigt trotzdem die Übersichtlichkeit des zu bearbeitenden Programmbildes nicht, Auf dem verhältnismäßig kleinen Bildschirmausschnitt kann ein Teil des Speichers in disassemblierter Form oder in hexadezimaler Schreibweise betrachtet werden. Weiterhin erfolgt noch die Ausgabe der Positionen der vier Breakpoints sowie der Prozessor- und Statusregister. Hier fehlen zwar Ein- und Ausgabefunktionen, dafür gibt es aber einige unübliche Bonbons. So kann man mit START das Programm an der Stelle fortsetzen, an der sich der PC (Program-Counter) befindet. SELECT erfolgt ein Umschalten zwischen dem DDT-Bildschirm und dem vom Programm erzeugten Bildschirm, und mit OPTION lassen sich noch einzelne Maschinensprachbefehle abar-

Eine lästige Eigenschaft von DDT soll hier nicht verschwiegen werden. Da der Speicherausschnitt im Bildschirmfenster nur mit den Cursortasten um einzelne Zeilen verschoben werden kann, ist es leider ein Unding, bestimmte Programmteile schnell zu finden.

Zusammenfassend kann man jedoch sagen: DDT ist zum Bearbeiten von Programmen, die sich bereits im Speicher befinden, prädestiniert.

Als Zusatz zum »MAC/65«-Assembler wird von OSS eine Diskette mit Hilfsprogrammen und vordefinierten Makros (»MAC/65 Toolkit«) angeboten. Hier findet man beispielsweise Makros für alle normalen Ein- und Ausgabefunktionen, normale und Player/Missile-Grafik und Arithmetik. Darunter befindet sich auch ein Programm, das die Kommandozeile von DOS XL analysiert

und somit das Schreiben von zusätzlichen Kommandos für DOS XL wesentlich erleichtert. Ein komplettes Dateikopierprogramm, das fast ausschließlich mit Makros geschrieben ist, fehlt auch nicht.

Man kann mit gutem Gewissen sagen, daß MAC/65 für jeden, der sich in Assembler versuchen will, hervorragend geeignet ist. Dabei kommen auch

Anfänger nicht zu kurz, die sich in der Basic-ähnlichen Umgebung wohlfühlen werden. Ein fortgeschrittener Programmierer wird nach einigen Stunden Arbeit mit »MAC/65« nicht mehr auf diesen komfortablen Assembler verzichten wollen. (Julian F. Reschke/wb)

Bezugsqueilen, Münzenloher, Tötzer Str. 4, 8150 Holzkirchen, Tel. (08024) 1814 Compy-Shop, Gneisensustr. 29, 4330 Mühlheim, Tel. (0208) 4971 69

# »SynFile+«, Dateiverwaltung total

Vor allem durch eine gelungene Benutzerführung und ein gut durchdachtes Konzept hebt sich »SynFile+« angenehm von anderen Dateiverwaltungen ab.

at man schon mal einen Heimcomputer zu Hause stehen, sollte man ihn nicht nur zum Spielen nutzen. Schließlich gibt es eine Reihe von sinnvollen Anwendungen, die sich auch auf einem Atari realisieren lassen. So zum Beispiel eine Dateiverwaltung wie »SynFile+« Mit ihr kann man Daten aller Art übersichtlich und sauber verwalten. Sei dies nun ein Adressenverzeichnis, eine Plattensammlung oder Mutters Kochrezepte, das spielt keine Rolle. Aber auch Kundendaten oder Verkaufsstatistiken kann man mit ein und demselben Programm verwalten.

»SynFile + «wird auf Diskette geliefert und ist für Atari Computer mit mindestens 48 KByte Speicher gedacht. Man erhält noch eine rund 120 Seiten umfassende Bedienungsanleitung in englischer Sprache, die jedoch durch Aufmachung und Gliederung gut zu handhaben ist. Auch die Benutzerführung des Programms ist natürlich in Englisch gehalten. Dies stört aber nicht so sehr, da die Bedienung von »SynFile+« vollständig menügesteuert ist.

Das Funktionsprinzip von »SynFile+« ist eigentlich ganz einfach. Am besten stellt man sich das Programm als Karteikasten vor. Jede Karteikarte, auf der man Daten eintragen kann, umfaßt 21 Zeilen mit je 80 Spalten. Auf dieser Karte trägt man zuerst die benötigten Datenfelder ein. Welche Felder man verwendet und wie man sie benennt. hängt natürlich vom eigenen Bedarf ab. Baut man sich beispielsweise ein Adressenverzeichnis auf, so würden die Felder »Name«, »Vorname«, »Wohnorts, >Telefons und so weiter lauten. Wie man diese Felder auf der Karte plaziert. bleibt dabei den eigenen Wünschen und Vorstellungen überlassen. Bei iedem Feld gibt man zudem noch die maximale Länge an, die jeder Eintrag umfassen darf. Außerdem muß der Typ des jeweiligen Feldes festgelegt werden. So soll ein Name beispielsweise später nur aus Buchstaben bestehen.

Andererseits setzt sich eine Telefonnummer aus numenschen Werten zusammen. Die Typenzuweisung birgt gleich zwei Vorteile in sich. Einmal kann das Programm dadurch leichter auf spätere Falscheingaben reagieren. Gibt man beispielsweise bei einer Telefonnummer versehentlich einen Buchstaben ein, so wird diese Eingabe aufgrund des nicht übereinstimmenden Feldtyps nicht akzeptiert. Zudem verfügt man bei ∍SynFile+« über eine Reihe von Feldtypen, die das Arbeiten mit dem Programm wesentlich komfortabler machen. So gibt es beispielsweise den Typ »Date«, also Datum. Hierbei sind nur Eingaben erlaubt, die tatsächlich auch einen Bezug zum Datum haben. Gibt man also »99/11/85 ein. reagiert »SynFile+« daraufhin mit elner Fehlermeldung. Allerdings muß ein Datum bei »SynFile+« in der amerikanischen Notierung, also erst der Monat, dann der Tag und zuletzt noch das Jahr. eingegeben werden.

Der Typ »Look-Up« wiederum erlaubt nur die Eingabe solcher Werte und Texte, die man vorher in einer Tabelle exakt festgelegt hat. Soll in einem Feld

Bild 1. So kann beispielswelse eine selbst aufgebaute Eingabemaske mit »SynFile+« aussehen

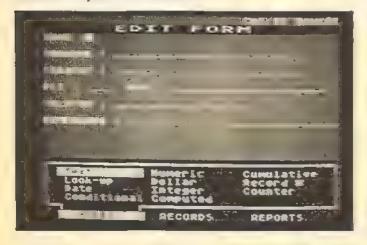


Bild 4. In allen Menús von »SynFile+« wählt man die Unterpunkte mit Hilfe der Cursortasten



NAME: Wurm VORNAME: Peter					
STRASSE: Petersstr. NR.: 11					
PLZ: 2222 ORT: Wurmhausen					
TELEFON: 2345					
GEBURTSTAG: 01/01/01 GESCHLECHT: M					
BEMERKUNG: wurmt					

#### Bild 2. Ein Datensatz als Hardcopy

zum Beispiel das Geschlecht einer Person eingetragen werden, so legt man in der Tabelle die Eingaben »m« für männlich und »w« für weiblich fest. Alle anderen Eingaben werden dann ignoriert. Mit Hilfe des Typs »Conditional« kann man dann noch einen Feldinhalt von einer vorher definierten Bedingung abhängig machen. Soll mit dem Inhalt eines oder mehrerer Felder gerechnet werden, kommt das Ergebnis dieser Berechnungen in ein Feld des Typs »Computed«.

Sobald die Definition aller Felder abgeschlossen ist, hat man eine fertige Eingabemaske, wie sie in Bild 1 zu sehen ist. Sollte man feststellen, daß die Maske noch nicht in der gewünschten Form vorliegt, kann man sie natürlich auch nachträglich noch ändern. Man ordnet der Datei einfach einen anderen Dateinamen zu und speichert sie auf Diskette. Als nächsten Schritt indiziert man ein oder mehrere Felder eines Datensatzes. Bei der Indizierung wird der Inhalt des jeweiligen Feldes in ein eigenes File abgelegt. Auf diese Weise muß das Programm beim Suchen oder Sortieren von Datensätzen nicht die gesamte Datei, sondern nur die jeweilige Indexdatei durchforsten. Bis zu 16 Felder können gleichzeitig indiziert werden. Dabei ist jedoch darauf zu achten, daß der Speicherplatz auf Diskette mit der Zahl der Indexfelder rapide abnimmt. Auch ist die Zahl der Datensätze, die eine Datei umfassen kann. vorwiegend durch die Größe des Indexfeldes begrenzt. Im Gegensatz zu den eigentlichen Daten muß das Indexfeld

GESAMTLISTE					
NAME	VORNAME	STRASSE	NR.	PLZ	ORT
Czerny	Wolfgang	Aschenstrasse	100	2222	Muenchen 12
Kohlrabi	Rotkohl	Weisskraut	1	4711	Sauerkraut
Markt	und	Technik	2	8013	Haar
Mueller	Hans	Dorfstrasse	10	1111	Gongdorf
Schraube	Nagel	Duebelstr.	2	2222	Holwurmhausen
Wurm	Peter	Petersstr.	11	2222	Wurmhausen

Bild 3. Eine Liste aller Datensätze läßt sich auf einem Drucker ausgeben

nämlich im Speicher des Computers Platz finden. Die Datensätze einer einzigen Datei hingegen können auf bis zu 16 Disketten abgelegt werden.

Hat man seine Daten eingegeben, kann man damit natürlich einige Manipulationen vornehmen. So lassen sich die Datensätze nach steigender oder fallender Folge sortieren. Einzelne Datensätze kann man auch mit Hilfe von verschiedenen Suchkriterien schnell finden. Gibt man mehrere Suchkriterien an, kann man wählen, ob alle Bedingungen gleichzeitig oder nur eines der Kriterien erfüllt sein muß.

Will man mit Hilfe einer Textverarbeitung Serienbriefe drucken, können die Daten, die in die Briefe eingefügt werden sollen, natürlich einer SynFile-Datei entnommen werden. Ein Beispiel: Man schickt ein Rundschreiben an alle Personen, die in einer Adreßdatei gespeichert sind und in einer bestimmten Stadt wohnen. Anschließend gibt man noch als Suchkriterium die gewünschte Stadt ein. Aus den gefundenen Datensätzen werden nun Namen und Adressen in ein separates File geschrieben. Eine Textverarbeitung wie der »Atari-Schreiber« beispielsweise, kann nun diese Datei lesen und deren Daten in die einzelnen Briefe eintragen.

Die Inhalte einer Datei kann man natürlich auch auf einem Drucker ausgeben. Dies kann grundsätzlich auf drei Arten geschehen. Einmal kann man quasi eine Hardcopy des momentan auf dem Bildschirm befindlichen Datensatzes anfertigen (Bild 2). Will man mehrere Datensätze ausdrucken, kann man die einzelnen Felder entweder nebeneinander oder untereinander darstellen (Bild 3). Auch beim Ausdruck kann man sich die jeweiligen Datensätze mit Hilfe von Suchkriterien beliebig zusammenstellen. Auf diese Weise lassen sich auch ohne weiteres Adreßetiketten beschriften. Leider sieht »SynFile+« nicht die Möglichkeit vor, eine Druckmaske zu speichern.

Die Benutzerführung von »SynFile+« ist als vorbildlich zu bezeichnen. Am unteren Bildrand findet man drei Hauptmenüpunkte. Mit Hilfe der Cursortasten wählt man sich den jeweiligen Punkt aus. Drückt man dann RETURN, so erscheint das jeweilige Untermenü In Form eines Fensters auf dem Bildschirm (Bild 4). Die einzelnen Untermenüpunkte werden wiederum mit den Cursortasten angewählt. Auch die Typenzuweisung bei der Felderdefinition wird über ein Untermenü durchgeführt. Mit diesem Bedienungsschema erreicht »SynFile+« eine Übersichtlichkeit, wie sie bei kaum einem Programm für Atari-Computer zu finden ist.

Mit Ausnahme der etwas mangelhaften Druckeranpassung kann man 
»SynFile+« also kaum Schwächen nachsagen. Ein echtes Kaufhindernis dürfte jedoch der Preis dieses Programms sein. Bei rund 220 Mark sollte man doch eine rentable Anwendung für dieses Programm vorweisen können. 
(Wolfgang Czerny/wb)

Bezugsquellen: CompyShop, Gneisensustr, 29, 4330 Mühlhelm/Ruhr, Tel. (0208) 497169 Münzenfoher, Tölzer Str. 4, 8150 Holzkirchen, Tel. (08024) 1814









#### J. Schultz/W. Pest Orucker-Handbuch Januar 1985, 188 Seiten

Welchen Drucker brauche ich? Einen schnellen, einen leisen, einen billigen, einen korrespondenzfähigen? Welcher Drucker kann problemios an meinen Personal- oder Homocomputer ange-schlossen werden? Wie funktioniert so

ein Drucker überhaupt? Lesen Sie dieses Buch, und Sie wissen auf alle diese Fragen eine fundierte Ant-

wort!

• Endlich ein informativer Leitfaden für alle, die vor dem Kauf eines Druckers stehen.

Bast-Nr. MT 742 ISBN 3-89090-077-1

DM 38-



#### J. Willis/M. Miller Computertechnik ohne Geheimnisse November 1984, 313 Seiten

Ob Sie schon einen Computer besitzen, seine Anschaftung planen oder sich nur mai aus interesse mit diesem faszinie ma aus interesse mit diesem raszine-renden Gebiet aussinandersetzen wol-len: in diesem Buch finden Sie die Ant-wort auf alle Fragen zur Computertech-nik. Das Buch beschreibt den derzeiti-gen Stand der Technik, gibt Aufschluß über die Entwicklung der Computer und derzeite haben des Aufschlussische verschafft Ihnen eine Marktübersicht Ober Hard- undSoftware (wenn Sie nicht wissen, wis das ist, dann lesen Sie die-ses Buchl). Und es beantwortet auf amusante Weise die oft gestellte Frage »Was, zum Kuckuck, kann ich eigentlich mit so einem Ding machen?«.

 Informationen zum Thema Computer von allgemeinem Interesse. Best.-Nr. M7 716 ISBN 3-89090-066-6

DM 42,-

Markt & Technik-Fachbücher erhalten Sie bei Ihrem Buchhändler.

Markt&Technik BUCHVERLAG

Haws-Piesel-Straße 2, 8013 Haar bei München

## Depot-Händler

Tragen Sie Ihre Buchbestellung auf eine Postkarte ein und schicken diese an einen Depothändler in Ihrer Nähe oder an Ihren Buchhändler.

ng Herder, Kurtürstendamm 69 15, Tel. (0:30) 583 5002. 5100 Acchen, 18t. (024) 481 42 Buchhendung Behrendt, Am Hof 52 5300 Bonn 1, 76t. (0228) 658021 Buchhendung Cusaruru, Schloßstraße 12 Buchhendung Cusaruru, Schloßstraße 12 Abad, Buchandung Interhabout, Florechstraße 61-65 5500 Tier, 101 (02511 43596 Buchhendung W. Finke, Ripdorf 32 5600 Wungparral 1, 76t. (1020) 424420 Buchhandlung Belagh, Sendstraße 1 5900 Bingen, 76t (10271) 55298-9 Buchhandlung Nascher, Steinweg 3 6000 Frankturt 1, 76t. (1058) 298050 Buchhandlung Welsen Holler, Schlosser 1864 6100 Dormstodt, 16t. (1059) 298050 Buchhandlung Welsen Holler, Schlosser 1864 6100 Dormstodt, 18t. (1054) 11 12091 Buchhandlung Selever Caches, Franktursße 31 6300 Gleißer, 78t. (1064) 1 12091 6400 Fulde, Tel. (1064) 1750 72 Cutrenberg Buchhandlung, Große Bielersweg 83 6500 Meist, 76t. (1064) 13 07 11 Buchhandlung Bock + Selb, Eutterstraße 2 6600 Saarbrucker, Tel. (1068) 306 77 Duchtmodung Welselm Hofmann, Bisrnarchstraße 98 6700 Ludwigshäfen, Tel. (1062) 3 16071 Buchhandlung Welselm Hofmann, Bisrnarchstraße 98 6700 Ludwigshäfen, Tel. (1062) 3 16071 Buchhandlung Welselm Hofmann, Bisrnarchstraße 98 6700 Ludwigshäfen, Tel. (1062) 3 16071 Buchhandlung Welselm Hofmann, Bisrnarchstraße 98 6700 Ludwigshäfen, Tel. (1062) 21 299 12 Buchhandlung Shibin, Bishrhoftstraße 13 7000 Shirtgart 150, Tol. (1071) 18 51 476 

Eicher Miore & Personal Computer, Hürstingen SS-SS 8-4780 St. Vith. Tel. (080) 22 73 93

Litrahle Premioulture, 14, no Duchacher PL de Perie) L-1011 Luxembourg-Gare, Tel. 480691, Telex 3112

Markt&Technik **BUCHVERLAC** 

### **Impressum**

Herausgeber: Carl-Franz von Quadt, Otner Weber

Chefredakteur: Michael Scharlenberger (ac) Leitender Redakteur: Michael Lang (ig) Redakteure: Werner Breuer (wb; Inhali) Petra Wanglor, Eva Hierimeier (Koordination) Redaktionsassistenz: Monika Lowandowski (222)

Fotografie: Jens Jancke

Layout Lee Eder (Lig.).

Signid Kowalewski (Cheflayouterin)

Auslandsrepräsentation:

Schweiz Markt & Technik Vertriebs AG. Kollerstrasse 3, CH-6300 Zug.

Tel. (042) 41 56 56, Telex: 8 62 329 mut ch

USA: M&T Publishing, 2464 Embarcadero Way, Palo Alto, CA 94303; Tel. 415-424-0600; Telex 752351

Manuskripteinsendungen: Manuskripte und Programm ästings werden gerne von der Redaktion angenommen. Sie müssen frei sein von Rechten Dritter, Sollten sie auch an anderer Stelle zur Veröffentlichung oder geworblichen Nutzung angeboten worden sein, muß dies angegeben werden. Mit der Einsendung von Manuskripten und Listings gibt der Verfasser die Zustimmung zum Abdruck In von der Markt & Technik Verlags AG herausgegebenen Publikationen und zur Vervielfättigung der Programm-lissings auf Derenträger. Mit der Einsandung von Bauanleitungen gibt der Einsender die Zustimmung zum Abdruck in von Markt & Tochnik Verlag AG verlegten Publikationen und dazu, daß Markt & Technik Verlag AG Geräte und Bautelle nach der Bauanleitung herstellen läßt und vertreibt oder durch Dritte vertreiben laßt. Honorare nach Vereinbarung. Für unverlangt eingesandte Manuskripte und Listings wird keine Haftung übernommen.

Produktionsleitung: Klaus Buck (180)

Anzeigenverkauf: Brigitta Fiebig (211)

Anzeigenverwaltung und Disposition:

Patricia Schiede (172)

Marketingleiter Vertrieb: Hans Hörl (114)

Vertriebsleitung: Helmut Grünfeldt (189)

Verlagsleiter M&T Buchverlag: Günther Frank (212)

Vertrieb Handelsauflage: Inland (Groß-, Einzel- und Bahnhotsbuchhandel) sowie Osterreich und Schweiz: Pegasus Buch- und Zeitschriften-Vertriebs GmbH, Hauptstätter Str. 96, 7000 Stuttgart 1, Tel. (07 11) 6483-0

Bezugsmöglichkeiten: Leser-Service: Telefon (089) 4613-249. Bestellungen nimmt der Verlag oder jede Suchhandlung entgegen.

Bezugspreis: Das Einzelheft kostet DM 14,-.

Druck: Druckhaus München GmbH, Schellingstraße 39-43, 8000 München 40

Urheberrecht: Alle in diesem Sonderheft erschlenenen Beiträge sind urheberrechtlich geschützt. Alle Rechte, auch Übersetzungen, vorbehalten. Reproduktionen gleich welcher Art, ob Fotokopie, Mikrofilm oder Erfassung in Datenverarbeitungsanlagen, nur mit achriftlicher Genehmigung des Verlages. Anfragen sind an Michael Scharfenberger zu richten. Für Schaltungen, Bauanleitungen und Programme, die als Beispiele veröffentlicht werden, können wir weder Gewähr noch irgendwelche Haftung übernehmen. Aus der Veröffentlichung kunn nicht geschlossen werden, daß die beschriebenen Lösungen oder verwendeten Bezeichnungen frei von gewerblichen Schutzrechten sind. Anfragen für Sonderdrucke sind an Peter Wagstyl zu richten

© 1986 Markt&Technik Verlag Aktiengesellschaft, Redaktion «Happy-Computer».

Verantwortlich: Für redaktionellen Teil Michael Scharfenberger

Für Anzeigen: Ralph Peter Rauchfuß (126).

Worstand: Carl-Franz von Quadt, Otmar Weber

Anschrift für Verlag, Redaktion, Vertrieb, Anzeigenverwaltung und alle Verantwortlichen: Markt&Technik Verlag Aktiengesellschaft, Hans-Pinsel-Straße 2, 8013 Haar bei München,

Telefon (089) 46 13-0, Telex 5-22052

#### Telefon-Durchwahl im Verlag:

Wählen Sie direkt: Per Durchwahl erreichen Sie alle Abteilungen direkt. Sie wählen 0.89/4613 und dann die Nummer, die in Klammern hinter dem leweitigen Namen angegeben ist.

Aktionare, die mehr als 25% des Kepitals halten Otmar Weber, Ingenieur, München; Carl-Franz von Quadt, Setriebswirl, München; Aufsichtsrat: Dr. Robert Disamann (Vorsitzender), Karl-Heinz Fanselow, Eduard Heilmayr



# **COMPUTER-ZEITSCHRIFTEN VON PROFIS FÜR PROFIS**

## **COMPUTER PERSÖNLICH**

Das aktuelle Fachmagazin für Personal-Computer.

- ★ Wenn Sie jetzt den Schritt vom Heim-Computer zur professionellen Anwendung eines Personal Computers planen
- ★ Wenn Sie beruflich oder privat bereits einen Personal Computer benutzen
- \* Wenn Sie selbst programmieren
- ★ Wenn Sie professionelle Hardund Softwaretests suchen
- ★ Wenn Sie Ihr eigenes System möglichst effizient einsetzen wollen

dann ist »Computer persönlich«, das aktuelle Fachmagazin für Personal Computer, genau Ihre Zeitschrift.

Die konsequente Ausrichtung auf professionelle Anwendungen bietet Ihnen alle wichtigen Informationen.

Von Profis für Profis!

»Computer persönlich« gibt es alle 14 Tage neu bei Ihrem Zeitschriftenhändler oder im Computer-Fachgeschäft.

## PC MAGAZIN

Einzige Wochenzeitung für Personal Computer im IBM-Standard.

Sie beschäftigen sich beruflich oder privat mit dem Einsatz und der Anwendung von Personal Computern?

Sie sind an aktuellen, professionellen Informationen über IBM-PCs, kompatible Systeme und deren professionellen Einsatz interessiert? Dann ist das PC Magazin genau auf Ihre persönlichen Bedürfnisse zugeschnitten.

Es wird von anerkannten und erfahrenen Fachjournalisten für professionelle Anwender und Fachleute geschrieben.

Es berichtet jede Woche ausschließlich über Computer im IBM-Standard und kompatible Systeme, über Hard- und Softwareneuheiten. Es bringt ausführliche Testberichte und gibt Ihnen wichtige Informationen über Netzwerke sowie die PC/Host-Verbindung.

Nur diese Spezialisierung ermöglicht eine gezielte Berichterstattung und bietet genügend Raum, um auf Anwenderprobleme spezifisch eingehen zu können.

Von Profis für Profis!

Und das jeden Mittwoch neu bei Ihrem Zeitschriftenhändler oder im Computer-Fachgeschäft.

#### GUTSCHEIN Sonden Sie mir die neueste Ausgabe der von mir angekreuzten Zeitschrift kostenios als Probeexemplar OMPUTER PERSÖNLICH Wenn mit Computer personisch zusagt und ich es regelmäßig welterbeziehen möchte, brauche ich nichts zu tun: Ich erhalte Wenn mir das PC-Magazin zusagt und ich es regelmäßig weiterbeziehen möchte, brauche lich nichts zu tun: Ich erhalte mein PC-Magazin brauche ich nichts zu tun: Ich ernatte Computer persönlich dann regelmaßig alle 14 Tage per Post frei Haus geliefert und bezahle pro Jahr nur DM 98,— Zuntellung dann regelmäßig jede Woche per Post frei Haus geliefert und bezahle pro Jahr nur DM 155,— Zustellung und Postgebühren übernimmt der Verlag. und Postochühren übernimmt der Verlag-Mir ist bokunnt, daß ich diese Gestellung innerhalb von 8 Tagen bei der Bestelladresse widerrufen Vorname/Name kann und bestätige dies durch meine zweite Unterschrift. Zur Wahrung der Frist genügt die rechtzeitige Absendung des Widerrufs. PLZ/Ort Straße Datum t. Unterschrift Datum 2. Unterschrift

Gutachein ausfüllen, ausschneiden, auf Postkarte klaben und einsenden an Markt&Technik Verlag Aktiengesellschaft, Vertrieb, Postfach 1304, 8013 Haar Markt & Technik
ATARI STSoftware



mit MailMerge für die ATARI ST-Computer

31/2"-Format

# WordStarfür den ATARI ST

Der Bestseller unter den Textverarbeitungsprogrammen bietet Ihnen bildschirmorientierte Formatierung, deutschen Zeichensatz und DIN-Tastatur sowie integrierte Hilfstexte. Mit MailMerge können Sie Serienbriefe mit persönlicher Anrede an eine beliebige Anzahl von Adressen schreiben und auch die AdreBaufkleber drucken.

Jetzt gibt gs WordStar/MailMerge für den ATARI ST! Damit eröffnen sich Ihnen alle Möglichkeiten, Ihren ATARI ST für professionelle Textverarbeitung einzusetzen. Zum Superpreis!

- CP/M-Z 80-Emulator
- WordStar/MailMerge-Dateien

Hardware-Anforderungen: ATARI ST-Computer, 80-Zeichen-Monitor, ein 31/2"-Diskettenlaufwerk, beliebiger Drucker mit Centronics-Schnittstelle.

WordStar ist an den ATARI ST bereits fertig angepaßt und läßt sich bequem über Funktionstasten steuern.

Bestell-Nr. MS 105

Für sagenhafte

DM 199,-"

(sFr. 178,-)

"inkl. MwSt. Unverbindliche Preiseungleitung

Markt & Technik-Softwareprodukte erhalten Sie in den Computer-Akteilungen der Kaufhäuser und im Computershop.

Wenn Sie direkt heim Verlag hestellen wellen: Ner per Nachnahme, gegen Vortuskasse, Verrechnungsscheck oder mit der eingehefteten Zahlkarte in diesem Heft. Besteflungen im Ausland: Schweiz: Markt & Technik Vertriebs AG, Koderstrasse 3, CH-6300 Zug, ©042/415656; Österreich: Ueberreuter Media Handels- und Verlagsges. mbil, Alser Straße 24, 1091 Wien, Tel. 0222/481538-8

Markt&Technik
Atari ST-Software

Hans-Pinsel-Straße 2, 8013 Haar bei München